LEARNING INCOMPRESSIBLE 3D FLOWS FROM SPARSE SAMPLES: A SET TRANSFORMER APPROACH TO DIVERGENCE-FREE KERNEL FITTING

Anonymous authors
Paper under double-blind review

ABSTRACT

Recovering divergence-free velocity fields from sparse and corrupted observations requires reconciling physical consistency, expressive representations, and robustness to missing data. We introduce **DFK-Former**, a set-to-field transformer that reconstructs divergence-free velocity fields from sparse, noisy, and irregularly sampled 3D observations. The model is inherently invariant to the number and ordering of input samples and predicts the parameters of a divergence-free kernel representation, producing continuous, mesh-free velocity fields that satisfy incompressibility exactly by construction rather than approximately through soft constraints or projection operators. Unlike existing divergence-free kernel approaches, which estimate kernel coefficients via a separate optimization for each field, DFK-Former predicts these coefficients directly in a single forward pass. This eliminates costly iterative fitting while providing a learned prior that enables accurate reconstruction even when observations are insufficient to uniquely determine the field. To further enhance representational power, we introduce adaptive and anisotropic divergence-free kernels that preserve exact incompressibility guarantees while capturing more complex flow structures. Extensive experiments across varying levels of sparsity, noise, and occlusion, including transfer to real Direct Numerical Simulation (DNS) turbulence data, demonstrate substantial gains in both accuracy and efficiency. DFK-Former reduces endpoint error by up to $4.6\times$ compared to per-field divergence-free-kernel optimization, while reconstructing each field in $7.5\ ms$ instead of $9.8\ s$, yielding a speedup of more than three orders of magnitude. These results establish direct, feed-forward prediction of divergence-free kernel parameters as a practical and effective paradigm for physically consistent velocity-field reconstruction from sparse observations.

1 INTRODUCTION

Reconstructing dense, physically plausible velocity fields is a fundamental challenge in experimental fluid dynamics and simulation-driven graphics, with applications ranging from flow characterization and numerical-model validation to data-driven simulation. Particle Imaging Velocimetry (PIV) and its variants measure flow by tracking passive tracers (Adrian & Westerweel, 2011; Scarano, 2013; Xiong et al., 2021; Wang et al., 2020). Alternative approaches infer velocity from light-transport measurements (Atcheson et al., 2008; Ji et al., 2013) or directly from RGB video using physical constraints (Eckert et al., 2019; Zang et al., 2020). Sensor-network approaches, in turn, infer flow fields from sparse spatial observations (Erichson et al., 2020; Nguyen et al., 2026). Regardless of the modality, the measurements are sparse, incomplete, and contaminated by noise, making recovery of the underlying velocity field a challenging task but ideally suited for machine learning.

In many fluid-flow settings, the underlying velocity field is approximately incompressible and should therefore have zero divergence (i.e., divergence-free). However, in practice, measured observations $\boldsymbol{u}_m$ deviate from this physical constraint due to noise, occlusions, limited spatial resolution, and interpolation artifacts. As a result, reconstructing the physically admissible flow involves estimating the divergence-free component of the observations, typically through the Leray (Helmholtz–Hodge) projection: $\boldsymbol{v}^* = P_{\textit{div-free}}(\boldsymbol{u})$. Although straightforward in a densely sampled setting, the problem becomes more challenging when observations are sparse and corrupted.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Recent approaches to divergence-free reconstruction address complementary aspects of this problem. Divergence-Free Kernels (DFKs) (Ni et al., 2025) represent velocity fields as linear combinations of compactly-supported matrix-valued basis functions obtained by applying differential operators to Wendland polynomial C^4 (Wendland, 1995). These kernels guaranty exact incompressibility by construction, while yielding mesh-free and continuously queryable velocity representations. However, DFKs are fitted on a per-instance basis. Consequently, every reconstruction requires the solution of a new optimization problem from scratch, incurring a high computational cost without learning prior knowledge about the distribution of physically plausible flows. In contrast, HDNet (Qi et al., 2024) learns a Helmholtz decomposition from data using a convolutional architecture, effectively capturing statistical regularities of flow fields through a learned prior. Yet its divergence-free output is only approximate, relying on discrete grid-based operations without exact guarantees, and the method assumes dense observations on a structured 2D grid. To the best of our knowledge, no existing method unifies learned flow priors with exact divergence-free guarantees for sparse, unstructured 3D observations.

Recent advances in permutation-equivariant architectures, including Set Transformer (Lee et al., 2019), Perceiver IO (Jaegle et al., 2022), and VGGT (Wang et al., 2025), have demonstrated the ability to infer structured geometric representations directly from unordered sets of observations in a single forward pass. Building on this idea, we propose **DFK-Former**, which couples a transformer-based set encoder with a divergence-free kernel representation. Given an unordered set of observations $\mathcal{O} = \{(\boldsymbol{p}_m, \boldsymbol{u}_m, c_m)\}_{m=1}^{M}$ consisting of 3D positions, noisy velocity measurements, and confidence scores, **DFK-Former** predicts the parameters (Θ) of a DFK representation $\boldsymbol{v}_\Theta \approx \boldsymbol{v}^*$. The resulting velocity field is exactly divergence-free by construction, remains mesh-free and continuously queryable, and can be reconstructed in a single forward pass without per-instance optimization at inference time. The main contributions of this work are as follows:

- We introduce **DFK-Former**, which combines learned flow priors with exact divergence-free guarantees for sparse, unstructured 3D observations by predicting kernel weights, locations, and support radii directly, without per-instance inference optimization.
- We develop anisotropic divergence-free kernels that capture coherent flow structures and improve reconstruction at matched model capacity.
- We extend the model to unsteady flows using alternating spatial and temporal attention with Fourier-in-time kernels, preserving exact incompressibility at every instant.
- We establish a comprehensive evaluation framework and demonstrate gains across sparsity, noise, and occlusion, including on real DNS data.

2 Related Work

Flow Reconstruction from Sparse Observations. Recovering dense velocity fields from incomplete and noisy measurements is a central problem in fluid dynamics and scientific machine learning. In experimental fluid mechanics, PIV and its extensions estimate flow fields by tracking tracer particles across image sequences (Adrian & Westerweel, 2011; Scarano, 2013; Wang et al., 2020; Xiong et al., 2021). Related approaches reconstruct fluid motion from light transport measurements, scalar transport observations, or monocular RGB videos (Atcheson et al., 2008; Ji et al., 2013; Eckert et al., 2019; Zang et al., 2020). Across modalities, measurements suffer from limited coverage, occlusions, artifacts, and noise. Classical remedies use interpolation, variational optimization, or data assimilation; recent learning-based methods reconstruct fields from sparse sensors, images, or scattered samples (Erichson et al., 2020; Nguyen et al., 2026). A closely related direction represents physical quantities using continuous coordinate-based neural fields, enabling mesh-free reconstruction from sparse observations (Sitzmann et al., 2020; Tancik et al., 2020). Applications include Neural Flow Maps (Sahoo et al., 2022) and Physics-Enhanced Implicit Neural Representations (Shen et al., 2025). However, these methods solve a separate optimization problem for each scene and enforce physical constraints only approximately. In contrast, our approach learns a direct mapping from sparse observations to velocity fields while targeting exact incompressibility.

Divergence-Free Representations and Physical Constraints. Many learning-based approaches incorporate physical consistency through regularization terms, projection operators, or physics-informed objectives (Raissi et al., 2019; Li et al., 2021). For incompressible flows, several methods

encourage divergence-free solutions via soft penalties, projection operators, or vector-potential parameterizations (Raissi et al., 2020; Cai et al., 2021; Tompson et al., 2017; Qi et al., 2024; Li et al., 2026; Chang et al., 2022; Lyu et al., 2024). A complementary line of work encodes incompressibility directly in the representation itself, either through divergence-free neural architectures based on differential-form parameterizations (Richter-Powell et al., 2022) or through divergence-free radial basis functions and matrix-valued kernels that guarantee analytically divergence-free vector fields over the continuous domain (Narcowich & Ward, 1994; Wendland, 1995). Building on this foundation, Divergence-Free Kernels (DFKs) (Ni et al., 2025) introduced a compactly supported, continuously queryable representation for incompressible velocity fields with exact physical guarantees. Subsequent studies have explored integrating DFKs into broader fluid reconstruction (Tao et al., 2026) and simulation (Xing et al., 2026) pipelines. Unlike prior DFK-based approaches, which solve an optimization problem independently for each scene, our method learns to predict kernel parameters directly from observations, combining exact incompressibility with feed-forward reconstruction.

Set-Based Geometric Inference and Operator Learning. Learning-based reconstruction replaces expensive per-instance optimization with a predictor trained across instances (Garnelo et al., 2018; Gordon et al., 2019; Cranmer et al., 2020; Marino et al., 2018), and has proven effective for inverse problems and implicit representations (Siahkoohi et al., 2021; Tancik et al., 2021). Since sparse flow measurements form unordered sets rather than structured grids, architectures that process them should be permutation-invariant or equivariant by construction. Deep Sets (Zaheer et al., 2017) established this paradigm, and attention-based architectures such as Set Transformer (Lee et al., 2019) and Perceiver IO (Jaegle et al., 2022) extended it to model interactions among set elements. Building on these ideas, recent transformers infer continuous fields or 3D structure from irregular observations, spanning scattered-data interpolation and operator learning (Ding et al., 2024; Hao et al., 2023; Wu et al., 2024), as well as feed-forward 3D reconstruction from sparse views (Wang et al., 2025). Similarly, operator-learning methods such as DeepONet (Lu et al., 2021) and Fourier Neural Operators (Li et al., 2021) learn mappings between function spaces and have achieved strong results in fluid modeling. Several approaches leverage these ideas for sparse-sensor flow reconstruction (Zhao et al., 2024; Zhang et al., 2025; Dang & Nguyen, 2026). These methods regress field values without building physical constraints into the representation, so incompressibility is satisfied only approximately. In contrast, **DFK-Former** infers, in one forward pass, the parameters of a continuous divergence-free kernel representation with a permutation-equivariant transformer.

3 Background: Divergence-Free Kernel Fields

Modeling incompressible flows requires velocity fields ($\boldsymbol{v} : \mathbb{R}^3 \to \mathbb{R}^3$) satisfying $\nabla \cdot \boldsymbol{v}(\boldsymbol{x}) = 0$. Rather than enforcing incompressibility through divergence penalties or projection, Divergence-Free Kernels (DFKs) (Ni et al., 2025) guarantee this constraint analytically by construction. The velocity field is expressed as a superposition of matrix-valued divergence-free kernels as follows:

$$\boldsymbol{v}(\boldsymbol{x}) = \sum_{k=1}^{K} \psi_{\text{iso}}(\boldsymbol{x} - \boldsymbol{x}_k; \sigma_k)\, \boldsymbol{\omega}_k, \tag{1}$$

where $\boldsymbol{x}_k \in \mathbb{R}^3$ is the kernel center, $\sigma_k > 0$ its support radius, and $\boldsymbol{\omega}_k \in \mathbb{R}^3$ its vector coefficient, which controls its contribution to the flow. Using the divergence-free operator $\mathcal{D} = -\mathbf{I}\Delta + \nabla\nabla^\top$ [1], we define the isotropic kernel using the normalized Wendland-$\mathcal{C}^4$ potential as for (Ni et al., 2025):

$$\psi_{\text{iso}}(\boldsymbol{x}; \sigma) = \mathcal{D}\,\phi\left(\frac{\|\boldsymbol{x}\|}{\sigma}\right), \qquad \phi(r) = \frac{(\max(1-r, 0))^6 (35r^2 + 18r + 3)}{3}. \tag{2}$$

Since each kernel satisfies $\nabla \cdot \psi_{\text{iso}} = \mathbf{0}$, any linear combination remains exactly divergence-free.

$$\boldsymbol{v}_\Theta(\boldsymbol{x}) = \bar{\boldsymbol{v}} + \sum_{k=1}^{K} \psi_{\text{iso}}(\boldsymbol{x} - \boldsymbol{x}_k; \sigma_k)\, \boldsymbol{\omega}_k. \tag{3}$$

The constant term $\bar{\boldsymbol{v}}$ captures global transport while preserving incompressibility. The resulting parameterization $\Theta = \{\bar{\boldsymbol{v}}\} \cup \{(\boldsymbol{x}_k, \sigma_k, \boldsymbol{\omega}_k)\}_{k=1}^{K}$ defines a continuous divergence-free velocity field

[1] The derivatives in $\mathcal{D}$ are taken with respect to $\boldsymbol{x}$.

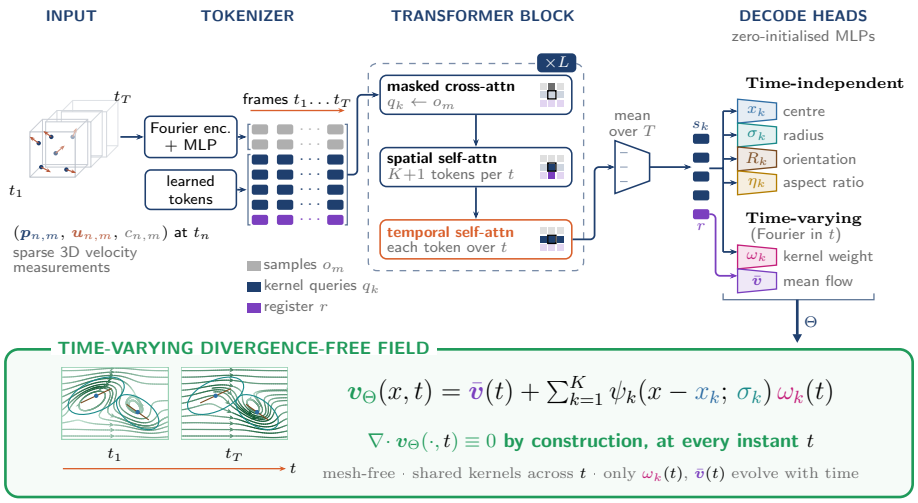

Figure 1: **DFK-Former architecture**. Sparse spatiotemporal velocity observations are encoded into tokens and jointly processed with learned kernel-query and register tokens using a Perceiver-style spatiotemporal transformer. Decoding heads predict the parameters of a divergence-free kernel representation, yielding a continuous time-varying incompressible velocity field by construction.

that can be evaluated at arbitrary query locations. In this work, DFKs serve as the structured output representation of our model: rather than predicting velocities directly, the network predicts the parameters Θ of the kernel field. This preserves the exact incompressibility guarantees of the DFK formulation while providing a compact and interpretable representation of the flow.

4 Method

Given a sparse, noisy, and unordered set of velocity observations $\mathcal{O} = \{(\boldsymbol{p}_m, \boldsymbol{u}_m, c_m)\}_{m=1}^{M}$, our goal is to reconstruct a continuous divergence-free velocity field. Instead of fitting a divergence-free kernel field independently for each observation set, we learn a direct mapping from observations to kernel parameters: $\mathcal{F}_\theta : \mathcal{O} \to \Theta$, where θ denotes the network parameters and Θ the DFK representation parameters introduced in Section 3. The predicted parameters are then evaluated through Eq. 3, guaranteeing incompressibility by construction. As illustrated in Fig. 1, DFK-Former combines a Perceiver-style transformer with a structured divergence-free kernel decoder. Rather than directly predicting velocities, the network predicts the parameters of a DFK field, yielding a physically interpretable latent representation in which each latent token corresponds to a kernel in the final velocity reconstruction.

4.1 DFK-Former Architecture

Observation Encoding and Kernel Queries. Each observation is embedded independently as $\boldsymbol{o}_m = \mathrm{MLP}\left([\gamma(\boldsymbol{p}_m) \| \boldsymbol{u}_m \| c_m]\right)$, where $\gamma(\cdot)$ denotes a Fourier positional encoding and c_m is an optional confidence indicator. In addition to observation tokens, the model maintains a set of K kernel-query tokens $\{\boldsymbol{q}_k\}_{k=1}^{K}$ and a global register token $\boldsymbol{r}$. Each query token is initialized from a learnable embedding together with the Fourier encoding of an initial kernel center $\boldsymbol{x}_k^0$, establishing an explicit correspondence between latent tokens and physical kernel locations. The transformer backbone alternates between cross-attention and self-attention, as shown in Fig. 1. During cross-attention, kernel-query tokens attend to the observation tokens and gather evidence from the sparse measurements. During self-attention, kernel and register tokens exchange information and coordinate their explanations of the observed flow. The resulting kernel-query states therefore summarize both local observations and global flow structure. Architecture hyperparameters are listed in Table 6.

Fixed-Kernel DFK Prediction. We first consider the DFK representation of Eq. 3 with fixed kernel locations and support radii. Kernel centers are initialized using Poisson-disk sampling over

the reconstruction domain (Ni et al., 2025), while support radii follow a predefined (optionally multi-scale) configuration. The prediction heads decode kernel coefficients and a global mean-flow component from the transformer outputs: $\boldsymbol{\omega}_k = f_\omega(\boldsymbol{q}_k)$ and $\bar{\boldsymbol{v}} = f_{\text{mean}}(\boldsymbol{r})$. The reconstructed field $\boldsymbol{v}_\Theta$ is then obtained from Eq. 3. This formulation replaces per-instance estimation of DFK coefficients with a learned predictor. For fixed kernel locations and support radii, the field is linear in $\{\boldsymbol{\omega}_k\}$ and $\bar{\boldsymbol{v}}$, and these parameters can be recovered by least-squares fitting under dense, clean observations. In the sparse, noisy, and partial-observation regime, however, the inverse problem becomes ill-conditioned. DFK-Former instead learns a data-driven prior that directly predicts the kernel coefficients and mean-flow term from observations in a single forward pass.

Adaptive Kernel Placement. Fixed kernel layouts cannot easily adapt to highly non-uniform flow structures. To increase flexibility, each kernel-query state directly predicts a kernel displacement and support radius, with $\boldsymbol{x}_k = \boldsymbol{x}_k^0 + \Delta\boldsymbol{x}_k$ and $\sigma_k = \sigma_k^0 \rho_k$. These quantities are decoded directly from the corresponding kernel-query state. The displacement and radius heads are initialized such that $\Delta\boldsymbol{x}_k = \mathbf{0}$ and $\rho_k = 1$, ensuring that the adaptive model initially behaves identically to the fixed-kernel formulation. Because DFKs remain divergence-free for arbitrary kernel locations and support radii, adaptive placement improves reconstruction capacity without sacrificing the physical guarantees of the underlying representation.

Anisotropic Divergence-Free Kernels. The kernels introduced so far are isotropic and therefore model flow structures using spherical regions of influence. However, many physically relevant structures, such as vortex tubes and shear layers, exhibit strong directional anisotropy. To capture such behavior, we replace isotropic supports with oriented ellipsoidal kernels. Each kernel is parameterized by a rotation matrix $\boldsymbol{R}_k \in \mathrm{SO}(3)$ and strictly positive physical semi-axis lengths $\boldsymbol{a}_k$. Let $\psi_{\text{ref}}(\boldsymbol{\xi}) := \psi_{\text{iso}}(\boldsymbol{\xi}; 1)$ denote the unit-radius kernel of Eq. 2. Following the Piola transform, we define the spatial profile of kernel k as follows:

$$\psi_k(\boldsymbol{z}) := \det(\boldsymbol{M}_k)^{-1} \boldsymbol{M}_k \psi_{\text{ref}}\left(\boldsymbol{M}_k{}^{-1}\boldsymbol{z}\right), \qquad \boldsymbol{M}_k = \boldsymbol{R}_k \,\mathrm{diag}(\boldsymbol{a}_k). \tag{4}$$

Eq. 4 is the contravariant Piola transformation associated with the linear map $\mathbf{z} = \boldsymbol{M}_k\boldsymbol{\xi}$. The Piola transformation commutes with the divergence operator and, therefore, maps divergence-free reference fields to divergence-free physical fields (Boffi et al., 2013). Consequently, $\nabla_{\mathbf{z}} \cdot [\psi_k(\mathbf{z})\boldsymbol{\omega}] = 0$ $\forall\, \boldsymbol{\omega} \in \mathbb{R}^3$, for any invertible $\boldsymbol{M}_k$. A direct derivation is provided in Appendix A. A key advantage of this construction is that divergence-freeness is preserved for any invertible transformation $\boldsymbol{M}_k$. Consequently, the network can learn kernel orientation and anisotropy directly from data while retaining exact incompressibility. In practice, we employ an axisymmetric parameterization $\boldsymbol{a}_k = (a_{\perp,k}, a_{\perp,k}, a_{\text{long},k})$. Under our extend-only convention, $a_{\perp,k} = \sigma_k$ and $a_{\text{long},k} = \sigma_k \eta_k$, where $\eta_k \geq 1$ is the aspect ratio. Thus $\boldsymbol{a}_k$ already includes the radius scale: no additional σ_k is applied to ψ_{ref}. For $\boldsymbol{R}_k = \mathbf{I}$ and $\boldsymbol{a}_k = \sigma_k(1,1,1)$, $\psi_k(\boldsymbol{z}) = \psi_{\text{iso}}(\boldsymbol{z}; \sigma_k)$, including its $\sigma_k{}^{-2}$ normalization. The anisotropic field uses $\psi_k(\boldsymbol{x} - \boldsymbol{x}_k)$ in place of $\psi_{\text{iso}}(\boldsymbol{x} - \boldsymbol{x}_k; \sigma_k)$ in Eq. 3.

4.2 Temporal DFK-Former

We extend DFK-Former to time-varying flows by operating on sequences $\{(\mathcal{O}_n, t_n)\}_{n=1}^T$, where n indexes frames, $t_n \in [0,1]$ is the normalized acquisition time, and $\mathcal{O}_n = \{(\boldsymbol{p}_{n,m}, \boldsymbol{u}_{n,m}, c_{n,m})\}_{m=1}^{M_n}$. The timestamp t_n is shared by all observations in frame n; the input tuple in Fig. 1 records this association. Observation embeddings use the same position, velocity, and confidence inputs as the static model; the timestamp is not an additional input to that observation MLP. We reserve $t \in [0,1]$ for continuous query time. Our key design principle is that kernel identities are preserved across time: the same K kernel-query tokens represent persistent flow structures throughout the sequence, eliminating the need for explicit kernel matching between frames. For each frame, observation tokens are computed as in the static model. Kernel queries attend to the observations through cross-attention and subsequently exchange information through alternating spatial and temporal self-attention. Spatial attention allows kernels within a frame to coordinate their explanations of the flow, while temporal attention propagates information across frames and promotes consistent kernel identities. The resulting per-frame kernel states are mean-pooled over time and decoded into Fourier

coefficients describing continuous coefficient trajectories $\boldsymbol{\omega}_k(t)$ and the mean-flow term $\bar{\boldsymbol{v}}(t)$:

$$\begin{bmatrix} \boldsymbol{\omega}_k(t) \\ \bar{\boldsymbol{v}}(t) \end{bmatrix} = \begin{bmatrix} \alpha_k^0 \\ \bar{\alpha}^0 \end{bmatrix} + \sum_{f=1}^{F} \left(\begin{bmatrix} \alpha_k^f \\ \bar{\alpha}^f \end{bmatrix} \cos(2\pi f t) + \begin{bmatrix} \beta_k^f \\ \bar{\beta}^f \end{bmatrix} \sin(2\pi f t) \right). \tag{5}$$

This low-order Fourier parameterization provides a compact continuous-time representation and naturally encourages temporal smoothness.

4.3 Procedural Training

Training Data Generation. All training examples are generated on the fly rather than drawn from a fixed dataset. For each training step, we sample a clean divergence-free target field $\boldsymbol{v}^*$ together with a corrupted observation set $\mathcal{O}$. This procedure prevents memorization of a finite training set and allows continuous variation of flow complexity, sampling density, and corruption severity. Our primary field generator samples Gaussian random fields on a periodic Fourier grid and projects them onto the divergence-free subspace using an exact Helmholtz decomposition in Fourier space,

$$\widehat{\boldsymbol{v}^*}(\boldsymbol{k}) = (I - \hat{\boldsymbol{k}}\hat{\boldsymbol{k}}^\top)\widehat{\boldsymbol{u}}(\boldsymbol{k}), \tag{6}$$

ensuring incompressibility to numerical precision (Appendix C, Fig. 3). Corrupted observations are generated by adding a controlled irrotational component and Gaussian noise, then applying sparse sampling and occlusion masks. The target remains the clean divergence-free field $\boldsymbol{v}^*$. To complement the spectral generator, we introduce a *Composite Analytic-Flow* (CAF) benchmark, which randomly combines multiple analytic flows, including Lamb–Oseen vortices, Taylor–Green vortices, ABC flows, shear layers, and vortex rings, with randomly sampled mixing weights. These flows introduce coherent structures that are difficult to obtain from Gaussian random fields alone.

Capacity-Matched Sampling. Since DFK-Former predicts a continuous field representation, supervision is performed at randomly sampled spatial query points rather than on a fixed grid. During training, we randomize the number of observations, sampling patterns, noise levels, and flow spectra to expose the model to a wide range of reconstruction conditions. To avoid introducing irreducible approximation error, the spatial spectrum of the generated fields is band-limited according to the representational capacity of the kernel basis (Appendix C.3). This ensures that training targets remain expressible by the chosen kernel configuration, allowing optimization to focus on inference from sparse observations rather than compensating for unattainable high-frequency content.

Training Objective. The primary supervision matches the reconstructed field at query locations, $\mathcal{L}_{\text{field}} = \mathbb{E}_{\boldsymbol{x}\sim Q}\left[\|\boldsymbol{v}_\Theta(\boldsymbol{x}) - \boldsymbol{v}^*(\boldsymbol{x})\|_2^2\right]$. In addition, we enforce consistency in the observed samples $\mathcal{L}_{\text{data}} = \frac{1}{M}\sum_{m=1}^{M} c_m \|\boldsymbol{v}_\Theta(\boldsymbol{p}_m) - \boldsymbol{v}^*(\boldsymbol{p}_m)\|_2^2$. Both terms are Huberized once strong corruption is introduced to reduce sensitivity to outliers. Two field-level regularizers further improve reconstruction quality. A magnitude-weighted cosine loss $\mathcal{L}_{\cos}$ improves directional accuracy, particularly under sparse observations, while an orthogonality loss $\mathcal{L}_{\text{ortho}}$ discourages alignment with the known irrotational contamination at observed samples. The cosine term is introduced gradually, and $\mathcal{L}_{\text{ortho}}$ is activated only under severe contamination. Additional losses regularize the geometry heads when enabled. For adaptive placement, $\mathcal{L}_{\text{ctr}}$ and $\mathcal{L}_{\text{rad}}$ keep predicted centers and radii close to their anchors, while a coverage term $\mathcal{L}_{\text{cov}}$ prevents kernel collapse by softly repelling centers. For anisotropic kernels, a warm-start loss $\mathcal{L}_{\text{sh}}$ initializes orientations and aspect ratios from sample-based estimates and is annealed to zero early in training. The final objective is a weighted sum of all these losses, with stage-specific weights provided in Table 7 and Table 9. Training follows a corruption curriculum, progressing from lightly corrupted observations to heavy noise, sparse sampling, and large occlusions.

Temporal Training. The temporal model is trained using procedurally generated time-varying divergence-free flows with two complementary generators. The first form trajectories by combining divergence-free spatial modes with time-varying Fourier coefficients, $\boldsymbol{v}(\boldsymbol{x}, t) = \sum_j c_j(t)\,\boldsymbol{v}_j(\boldsymbol{x})$, while the second models transported coherent structures through collections of advected divergence-free primitives (Appendix C.8). Both preserve incompressibility for all t. Observation counts, noise realizations, and occlusion masks are sampled independently for each frame. We supervise in randomly sampled space-time query locations using $\mathcal{L}_{\text{field}} = \mathbb{E}_{(\boldsymbol{x},t)}\left[\|\boldsymbol{v}_\Theta(\boldsymbol{x}, t) - \boldsymbol{v}^*(\boldsymbol{x}, t)\|_2^2\right]$, with the corresponding observation-consistency loss evaluated in the sampled measurements. Because Temporal DFK-Former represents kernel coefficients using a Fourier expansion of order F (Sec. 4.2),

the training data's temporal spectrum is band-limited to match model capacity. This avoids supervising temporal frequencies that the chosen Fourier basis cannot represent and enables a controlled study of the trade-off between temporal bandwidth and reconstruction accuracy.

5 RESULTS

5.1 EXPERIMENTAL SETUP

Datasets. We evaluate DFK-Former on 582 synthetic and real incompressible flow fields, drawn from four groups: (1) Gaussian Random fields, (2) the CAF set from Section 4.3 (drawn from the training distribution), (3) five analytic families with closed-form ground truth (Taylor–Green, Lamb–Oseen, vortex ring, shear layer, ABC), and (4) four real-DNS JHTDB sets (Li et al., 2008) (Isotropic turbulence, Forced MHD turbulence, Channel Flow, and Transitional Boundary Layer). Inputs are corrupted with varying sparsity ($M \in [187, 4086]$), noise ($\mathrm{RMS} \in [0, 15\%]$), irrotational contamination ($\chi \in [0, 1]$), and occlusion (0–35%), while targets remain clean Leray-projected velocity fields. For temporal reconstruction, we evaluated on real DNS sequences, comprising 60 time-resolved JHTDB flows with three unseen midpoint frames per sequence.
Metrics. Performance is evaluated through reconstruction accuracy (EPE, relative L^2, MAE, cosine similarity, angular error), physical consistency (divergence), and computational efficiency (latency, throughput, memory, FLOPs, and parameter count).
Baselines. We compare DFK-Former against optimization-based methods (Wendland RBF scalar interpolation, SIREN/Curl-SIREN (Sitzmann et al., 2020), vector-potential INRs (Richter-Powell et al., 2022), and DFK fitting (Ni et al., 2025)), operator-learning approaches (FNO and DeepONet), and physics-constrained models (PINNs with soft divergence penalties, and HDNet-3D [2]). We report two DFK variants: *fixed-kernels*, which fits only kernel weights on a predefined kernel layout, and *adaptive-kernels*, which jointly optimizes kernel positions, radii, and weights for each field. For the temporal extension, we compare against Space-Time (ST-) SIREN/Curl-SIREN, per-frame DFK fitting, and the shared-kernel sequence fitting approach proposed by Ni et al. (2025). Implementation details are given in Appendix B.

5.2 MAIN BENCHMARK

Table 1: Main reconstruction benchmark. Reconstruction accuracy, divergence residual, and per-field inference cost of DFK-Former against optimization-based, operator-learning, and physics-constrained baselines. Best in **bold**, second-best underlined.

Method	RMSE/EPE ↓	rel. L^2 ↓	MAE ↓	cosine ↑	angle (°) ↓	$\|\nabla\cdot v\|_2$ ↓	ms/field ↓	div-free?
Wendland RBF (scalar)	0.598	0.736	0.280	0.663	37.9	3.2	26	no
SIREN (+ soft penalty)	0.392	0.519	0.195	0.805	26.7	1.4	53600	no (soft)
Curl-SIREN	0.950	1.352	0.475	0.629	42.0	$\sim 10^{-14}$	52700	by construction
Vector-potential INR	0.778	1.092	0.388	0.661	39.1	$\sim 10^{-15}$	48000	by construction
DFK fitting (fixed kernels)	0.689	0.924	0.335	0.482	52.9	$\sim 10^{-15}$	22	by construction
DFK fitting (adaptive kernels)	0.546	0.719	0.264	0.568	45.5	$\sim 10^{-15}$	9821	by construction
FNO	0.244	0.312	0.121	0.900	16.9	0.71	4.3	no
FNO (+ Leray projection)	0.251	0.318	0.124	0.896	17.4	$\sim 10^{-16}$	4.9	by construction
DeepONet	0.417	0.509	0.203	0.764	30.5	0.23	**0.5**	no
PINN (+ soft penalty)	0.319	0.422	0.157	0.838	22.6	0.44	51000	no (soft)
HDNet-3D	<u>0.211</u>	<u>0.257</u>	<u>0.105</u>	<u>0.935</u>	<u>13.7</u>	0.66	<u>4.0</u>	partially
DFK-Former (ours)	**0.120**	**0.151**	**0.059**	**0.964**	**8.0**	$\sim 10^{-7}$	7.5	by construction

Table 1 compares all methods on the 582-flow benchmark in terms of reconstruction accuracy, divergence preservation, and inference cost. Optimization-based DFK fitting highlights the accuracy-computation trade-off: fixed-kernel fitting[3] is constrained by a predefined basis, whereas adaptive-kernel fitting improves accuracy through per-field optimization at substantially higher cost. Exactly divergence-free INRs such as Curl-SIREN and vector-potential models preserve incompressibility but remain challenging to fit from sparse and corrupted observations. Operator-learning baselines achieve stronger reconstruction accuracy, with HDNet-3D emerging as the strongest prior method and FNO offering a favorable accuracy-speed trade-off. Applying a Leray projection to FNO (and similarly to

[2] We use a 3D version of HDNet (Qi et al., 2024), trained on the same dataset as DFK-Former.
[3] Equivalent to an L^2 least-squares fit over a fixed kernel basis.

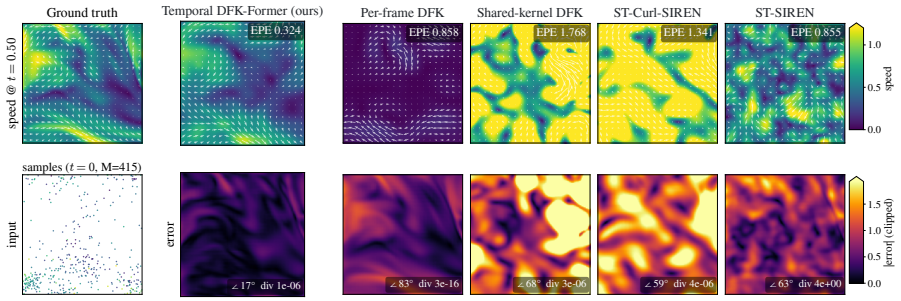

Figure 2: **Temporal reconstruction in the hardest evaluation regime** (Forced MHD (dynamic JHTDB), `hard combo`: sparse observations, noise, occlusions, and outliers). Top: ground truth and predictions from Temporal DFK-Former, per-frame DFK fitting, iterative DFK fitting, space-time Curl-SIREN, and space-time SIREN. Bottom: input observations and reconstruction errors. DFK-Former remains robust under severe degradation, recovering the dominant flow structures with substantially lower EPE and angular error than competing methods.

DeepONet) eliminates divergence but does not improve reconstruction accuracy. Thus, enforcing incompressibility after prediction cannot recover errors already present in the velocity field. While HDNet-3D projects reconstructed flows toward a divergence-free space, discretization on a voxel grid still introduces residual numerical divergence. DFK-Former achieves the best performance across all reconstruction metrics. Despite enforcing incompressibility to numerical precision ($\sim 10^{-7}$), it requires only 7.5 ms per field, comparable to fast feed-forward operator-learning models and roughly three orders of magnitude faster than optimization-based divergence-free approaches. These results demonstrate that directly predicting adaptive DFK parameters combines the accuracy and physical guarantees of divergence-free representations with the efficiency of feed-forward inference.

5.3 Time-varying flow

Table 2 shows that the temporal extension of DFK-Former achieves the lowest overall median EPE (0.114), outperforming per-frame DFK fitting (0.767), shared-kernel DFK fitting (0.626), and Space-Time SIREN/Curl-SIREN (0.428/0.842), while preserving incompressibility to numerical precision. It also achieves the lowest mean EPE (0.124) and mean angular error ($4.8°$). Gains are particularly pronounced under sparse and occluded observations, where temporal coherence substantially improves reconstruction quality, reducing error by up to a factor of 9.7 compared to per-frame fitting. Figure 2 illustrates the most challenging regime with combined sparse, occluded, and corrupted settings. Notably, the mean error of per-frame DFK fitting is heavily affected by a small number of near-singular optimization failures. Additional regimes are reported in Appendix F.6.

Table 2: Temporal flow reconstruction on 60 time-resolved JHTDB flows. Median EPE by observation regime, plus overall mean EPE and angular error; lower is better. Best results are in **bold**, second-best underlined.

Regime	**Temporal DFK-Former (ours)**	Per-frame DFK	Shared-kernel DFK	ST-SIREN	ST-Curl-SIREN
`clean&dense`	0.114	<u>0.100</u>	**0.083**	0.442	0.781
`sparse`	**0.109**	0.847	1.093	<u>0.388</u>	0.590
`occluded`	**0.098**	0.951	1.511	<u>0.413</u>	0.766
Overall (median EPE)	**0.114**	0.767	0.626	0.428	0.842
Overall (mean EPE)	**0.124**	20.29†	0.762	0.466	0.821
Overall (mean angular error)	**4.8°**	40.0°	25.4°	21.3°	37.1°

†Per-frame DFK's mean is inflated by $15/180$ near-singular per-frame solves.

5.4 Ablations

Representation vs. Backbone. To isolate the role of the DFK representation, we replace the DFK parameter head with a decoder that directly predicts velocity values while keeping the backbone and training protocol unchanged. The direct regressor achieves similar accuracy to DFK-Former

(EPE 0.127 vs. 0.120) at comparable latency (6.8 vs. 7.5 ms), but produces substantial divergence ($\|\nabla\cdot v\|_2 = 0.26$). This indicates that reconstruction accuracy primarily stems from the learned prior, while the DFK output space enforces incompressibility without sacrificing accuracy. Projecting the regressed field into the same DFK basis restores exact divergence-free structure but increases error (EPE 0.165) and runtime (29 ms), highlighting the benefit of directly predicting DFK parameters.

The Role of Attention. Replacing the attention-based encoder with permutation-invariant DeepSets (Zaheer et al., 2017) pooling while keeping all other components fixed significantly degrades reconstruction quality (EPE $0.558 \to 0.832$, angular error $42° \to 76°$). Although evaluated in a reduced-scale setting, this result suggests that modeling interactions between observations is essential, whereas simple symmetric pooling is insufficient despite preserving permutation invariance.

Representation Ladder Analysis. Table 3 presents cumulative ablations of the kernel parameterization. Adaptive placement yields modest gains, whereas multi-scale radii provide the largest benefit, reducing CAF EPE from 0.171 to 0.102 and improving performance across all datasets.

Kernel Count. Table 4 varies the number of kernels K while keeping all other settings fixed. Accuracy improves from $K = 32$ to 1024 (EPE $0.281 \to 0.120$), then degrades for larger budgets (0.144 at $K = 2048$). An independent $K = 8192$ run achieves a similar error, suggesting limited benefit from further increasing basis resolution. Inference cost remains nearly constant across a $32\times$ range of K since only the query tokens scale with the kernel count. Notably, even the smallest model ($K = 32$) outperforms closed-form DFK fitting, indicating that the gains arise primarily from the learned prior rather than basis size alone.

Anisotropic Kernels. To isolate representational gains from parameter-count effects, we compare the anisotropic model ($K = 1024$) with a parameter-matched isotropic variant ($K = 1542$). Table 5 shows consistent improvements across all analytic flows, with gains increasing from $1.16\times$ on Lamb–Oseen vortices to $1.34\times$ on vortex rings.

Table 3: **Component ladder.** Progressively adding adaptive placement and multi-scale radii to the anisotropic base model. Multi-scale radii provide the largest gains, particularly on CAF flows and real DNS data. Lower is better.

	in-dist		CAF		JHTDB (real DNS)	
Configuration	EPE↓	ang.°↓	EPE↓	ang.°↓	EPE↓	ang.°↓
Anisotropic (base)	0.382	28.3	0.164	11.4	0.167	9.3
+ adaptive placement	0.368	27.4	0.171	12.2	0.162	9.1
+ multi-scale radii	**0.343**	**25.5**	**0.102**	**6.9**	**0.155**	**8.7**

Table 4: **Kernel-Count Ablation.** Scaling the number of kernels improves reconstruction quality up to K=1024, which achieves the best accuracy-latency trade-off. Lower is better.

K	32	64	128	256	512	1024	2048
EPE↓	0.281	0.226	0.200	0.193	0.142	**0.120**	0.144
angle (°)↓	20.8	16.1	14.0	13.4	9.6	**8.0**	9.7
ms/field↓	**5.7**	**5.7**	5.9	5.9	6.0	7.5	11.6
peak mem (MB)↓	**58**	**58**	59	59	72	72	80

Table 5: **Anisotropy vs. a parameter-matched isotropic control** on structured analytic flows: our anisotropic K=1024 model against a separately trained isotropic model at K=1542, matched to within four scalar parameters. relEPE, lower is better; improvement is the isotropic/anisotropic ratio.

Target structure	Isotropic	Anisotropic	improvement
Vortex tube (Lamb–Oseen)	0.0763	**0.0656**	1.16×
Shear layer	0.0719	**0.0592**	1.22×
Vortex ring	0.0835	**0.0623**	1.34×

6 Conclusion and Limitations

We presented **DFK-Former**, a set-to-field transformer that predicts the parameters of a divergence-free kernel representation from sparse, noisy, and irregular observations. The resulting velocity fields are exactly divergence-free, mesh-free, and continuously queryable by construction, while avoiding per-instance optimization at inference time. Experiments across diverse reconstruction settings demonstrate that direct prediction of divergence-free kernel parameters combines the accuracy and physical guarantees of kernel-based representations with the efficiency and robustness of learned flow priors. Anisotropic kernels and temporal modeling further extend the framework to more complex spatial and temporal flow structures while preserving exact incompressibility.

Limitations include the use of synthetic training data and the limited exploration of larger model and kernel capacities. Although the learned representation generalizes to DNS turbulence without retraining, broader evaluation on experimental measurements and larger-scale studies are needed to better characterize robustness and scaling behavior.

AI USE STATEMENT

In this work, we used large language models (LLMs) for four main tasks, all under author supervision.

Retrieval and discovery: to suggest candidate related work; every resulting reference was then located, read, and verified against the primary source by the authors, and any suggestion that could not be verified was discarded.

Research ideation and execution: to brainstorm design alternatives, and as a coding assistant for implementing and debugging the model, baselines, data generators, and evaluation process, as well as for summarizing the resulting experiment logs.

Drafting: to produce first drafts of expository passages, which the authors then restructured, rewrote, and fact-checked against the underlying results.

Writing: to polish the grammar, phrasing, and clarity of the paper text, as well as to shorten some paragraphs when needed.

The feed-forward divergence-free kernel formulation and overall research direction are the authors' contributions. The outputs of LLMs were reviewed, corrected, and approved by the authors. All reported results were generated by executing the codebase and are supported by experiment logs and checkpoints; no numerical result, table, or figure was generated or estimated by an LLM. The codebase is covered by automated tests, and the authors take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

Code, Training and Evaluation Datasets. Upon acceptance, we will release the complete codebase, including the DFK kernel layer, DFK-Former, training and evaluation pipelines, procedural data generators, configurations, and scripts used to reproduce all reported results. Training data are generated on the fly from released procedural generators and corruption pipelines using fully specified configurations and random seeds. Synthetic evaluation sets are deterministically generated and distributed with accompanying metadata to ensure identical benchmarking across methods. For real-data experiments, we provide scripts and manifests that reproduce the exact JHTDB (Li et al., 2008) evaluation protocol for users with valid database access. Since JHTDB is a third-party resource, we do not redistribute the data; access requires an individual authorization token issued by the database. The released scripts and metadata enable users with valid credentials to reconstruct the exact evaluation datasets and preprocessing pipeline used in our experiments.

Environment. All experiments were run with PyTorch on NVIDIA A100 GPUs; the released repository pins the library versions and provides the environment specification, the checkpoints required to reproduce the reported evaluations, and the commands used to launch each run.

ACKNOWLEDGMENTS

Omitted for double-blind review.

REFERENCES

Ronald J Adrian and Jerry Westerweel. *Particle image velocimetry*. Number 30. 2011.

Bradley Atcheson, Ivo Ihrke, Wolfgang Heidrich, Art Tevs, Derek Bradley, Marcus Magnor, and Hans-Peter Seidel. Time-resolved 3D capture of non-stationary gas flows. *ACM TOG*, 27(5):1–9, 2008.

Daniele Boffi, Franco Brezzi, and Michel Fortin. *Mixed Finite Element Methods and Applications*. Springer Series in Computational Mathematics. Springer, 2013. doi: 10.1007/978-3-642-36519-5.

Robert Bridson. Fast poisson disk sampling in arbitrary dimensions. *ACM SIGGRAPH Sketches*, 2007.

Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (PINNs) for fluid mechanics: A review. *Acta Mechanica Sinica*, 37(12): 1727–1738, 2021.

Jumyung Chang, Ruben Partono, Vinicius C. Azevedo, and Christopher Batty. Curl-flow: Boundary-respecting pointwise incompressible velocity interpolation for grid-based fluids. *ACM TOG*, 41(6), 2022.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Hiep Vo Dang and Phong CH Nguyen. Deep operator learning for high-fidelity fluid flow field reconstruction from sparse sensor measurements. *J. Comput. Inf. Sci. Eng.*, 26(1):011007, 2026.

Shizhe Ding, Boyang Xia, Milong Ren, and Dongbo Bu. [niert: Accurate] numerical interpolation through unifying scattered data representations using transformer encoder. *IEEE Trans. Knowl. Data Eng.*, 36(11):6731–6744, 2024.

Marie-Lena Eckert, Kiwon Um, and Nils Thuerey. ScalarFlow: A large-scale volumetric data set of real-world scalar transport flows for computer animation and machine learning. *ACM TOG*, 38(6), 2019.

N Benjamin Erichson, Lionel Mathelin, Zhewei Yao, Steven L Brunton, Michael W Mahoney, and J Nathan Kutz. Shallow neural networks for fluid flow reconstruction with limited sensors. *Proceedings of the Royal Society A*, 476(2238):20200097, 2020.

Marta Garnelo, Dan Rosenbaum, Christopher Maddison, Tiago Ramalho, David Saxton, Murray Shanahan, Yee Whye Teh, Danilo Rezende, and S. M. Ali Eslami. Conditional neural processes. In *ICML*, volume 80, pp. 1704–1713, 2018.

Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E. Turner. Meta-learning probabilistic inference for prediction. In *ICLR*, 2019.

Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. GNOT: A general neural operator transformer for operator learning. In *ICML*, volume 202, pp. 12556–12569, 2023.

Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, et al. Perceiver IO: A general architecture for structured inputs & outputs. In *ICLR*, 2022.

Yu Ji, Jinwei Ye, and Jingyi Yu. Reconstructing gas flows using light-path approximation. In *CVPR*, pp. 2507–2514, 2013.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set transformer: A framework for attention-based permutation-invariant neural networks. In *ICML*, volume 97, pp. 3744–3753, 2019.

Xigui Li, Hongwei Zhang, Ruoxi Jiang, Deshu Chen, Chensen Lin, Limei Han, Yuan Qi, Xin Guo, and Yuan Cheng. Project and generate: Divergence-free neural operators for incompressible flows. *arXiv preprint arXiv:2603.24500*, 2026.

Yi Li, Eric Perlman, Minping Wan, Yunke Yang, Charles Meneveau, Randal Burns, Shiyi Chen, Alexander Szalay, and Gregory Eyink. A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence. *Journal of Turbulence*, (9):N31, 2008.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *ICLR*, 2021.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nat. Mach. Intell.*, 3(3):218–229, 2021.

Luan Lyu, Xiaohua Ren, Wei Cao, Jian Zhu, Enhua Wu, and Zhi-Xin Yang. Wavelet potentials: An efficient potential recovery technique for pointwise incompressible fluids. In *Comput. Graph. Forum*, volume 43, pp. e15023, 2024.

Joe Marino, Yisong Yue, and Stephan Mandt. Iterative amortized inference. In *ICML*, volume 80, pp. 3400–3409, 2018.

Francis J Narcowich and Joseph D Ward. Generalized hermite interpolation via matrix-valued conditionally positive definite functions. *Mathematics of Computation*, 63(208):661–687, 1994.

Phong CH Nguyen, Joseph B Choi, and Quang-Trung Luu. Deep learning with fourier features for regressive flow field reconstruction from sparse sensor measurements. *Scientific Reports*, 16(1): 5980, 2026.

Xingyu Ni, Jingrui Xing, Xingqiao Li, Bin Wang, and Baoquan Chen. Representing flow fields with divergence-free kernels for reconstruction. *Proc. ACM Comput. Graph. Interact. Tech.*, 8(4): 52:1–52:21, 2025.

Miao Qi, Ramzi Idoughi, and Wolfgang Heidrich. Hdnet: physics-inspired neural network for flow estimation based on helmholtz decomposition. *arXiv preprint arXiv:2406.08570*, 2024.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *J. Comput. Phys.*, 378:686–707, 2019.

Maziar Raissi, Alireza Yazdani, and George Em Karniadakis. Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations. *Science*, 367(6481):1026–1030, 2020.

Jack Richter-Powell, Yaron Lipman, and Ricky T. Q. Chen. Neural conservation laws: A divergence-free perspective. *NeurIPS*, 35:38075–38088, 2022.

Marie E. Rognes, Robert C. Kirby, and Anders Logg. Efficient assembly of $h(\mathrm{div})$ and $h(\mathrm{curl})$ conforming finite elements. *SIAM Journal on Scientific Computing*, 31(6):4130–4151, 2009. doi: 10.1137/08073901X.

Saroj Sahoo, Yuzhe Lu, and Matthew Berger. Neural flow map reconstruction. In *Comput. Graph. Forum*, volume 41, pp. 391–402, 2022.

Fulvio Scarano. Tomographic PIV: principles and practice. *Measurement Science and Technology*, 24(1):012001, 2013.

Liming Shen, Liang Deng, Chongke Bi, Yu Wang, Xinhai Chen, Yueqing Wang, and Jie Liu. PEINR: A physics-enhanced implicit neural representation for high-fidelity flow field reconstruction. In *ICML*, 2025.

Ali Siahkoohi, Gabrio Rizzuti, Mathias Louboutin, Philipp A Witte, and Felix J Herrmann. Preconditioned training of normalizing flows for variational inference in inverse problems. *arXiv preprint arXiv:2101.03709*, 2021.

Vincent Sitzmann, Julien Martel, Alexander Bergman, David Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020.

Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. In *NeurIPS*, 2020.

Matthew Tancik, Ben Mildenhall, Terrance Wang, Divi Schmidt, Pratul P Srinivasan, Jonathan T Barron, and Ren Ng. Learned initializations for optimizing coordinate-based neural representations. In *CVPR*, pp. 2846–2855, 2021.

Ningxiao Tao, Baoquan Chen, and Mengyu Chu. LagrangianSplats: Divergence-free transport of gaussian primitives for fluid reconstruction. In *SIGGRAPH*, pp. 52:1–52:12, 2026.

Jonathan Tompson, Kristofer Schlachter, Pablo Sprechmann, and Ken Perlin. Accelerating eulerian fluid simulation with convolutional networks. In *International Conference on Machine Learning*, pp. 3424–3433, 2017.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotný. VGGT: visual geometry grounded transformer. In *CVPR*, pp. 5294–5306, 2025.

Yuanhao Wang, Ramzi Idoughi, and Wolfgang Heidrich. Stereo event-based particle tracking velocimetry for 3D fluid flow reconstruction. In *ECCV*, pp. 36–53, 2020.

Holger Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Advances in computational Mathematics*, 4(1):389–396, 1995.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. In *ICML*, volume 235, pp. 53681–53705, 2024.

Jingrui Xing, Yizao Tang, Mengyu Chu, and Baoquan Chen. DDFKs: Fluid simulation with dynamic divergence-free kernels. *arXiv preprint arXiv:2601.17693*, 2026.

Jinhui Xiong, Andres Alejandro Aguirre-Pablo, Ramzi Idoughi, Sigurdur T Thoroddsen, and Wolfgang Heidrich. RainbowPIV with improved depth resolution—design and comparative study with TomoPIV. *Measurement Science and Technology*, 32(2):025401, 2021.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. *Advances in Neural Information Processing Systems*, 30, 2017.

Guangming Zang, Ramzi Idoughi, Congli Wang, Anthony Bennett, Jianguo Du, Scott Skeen, William L Roberts, Peter Wonka, and Wolfgang Heidrich. TomoFluid: Reconstructing dynamic fluid from sparse view videos. In *CVPR*, pp. 1870–1879, 2020.

Qian Zhang, Dmitry Krotov, and George Em Karniadakis. Operator learning for reconstructing flow fields from sparse measurements: An energy transformer approach. *J. Comput. Phys.*, 538:114148, 2025.

Xiaoyu Zhao, Xiaoqian Chen, Zhiqiang Gong, Weien Zhou, Wen Yao, and Yunyang Zhang. RecFNO: A resolution-invariant flow and heat field reconstruction method from sparse observations via fourier neural operator. *Int. J. Therm. Sci.*, 195:108619, 2024.

Appendix: Supplementary Material

A Divergence Preservation under the Piola Transformation

The anisotropic kernel of Eq. 4 is a contravariant Piola transform of the isotropic kernel, the standard divergence-preserving map used to build $H(\mathrm{div})$-conforming finite elements (Boffi et al., 2013; Rognes et al., 2009). We restate the identity and verify it directly for our construction.

Proposition. Let $\psi_{\mathrm{ref}}(\boldsymbol{\xi}) := \psi_{\mathrm{iso}}(\boldsymbol{\xi}; 1) \in \mathbb{R}^{3\times 3}$ be the unit-radius kernel of Eq. 2, which is divergence-free in every column, and fix a kernel index k. Let $\boldsymbol{M}_k \in \mathbb{R}^{3\times 3}$ with $\det(\boldsymbol{M}_k) > 0$ be invertible. Define

$$\psi_k(\boldsymbol{z}) = \frac{1}{\det(\boldsymbol{M}_k)}\, \boldsymbol{M}_k\, \psi_{\mathrm{ref}}\big(\boldsymbol{M}_k{}^{-1}\boldsymbol{z}\big)\,. \tag{7}$$

Then ψ_k is divergence-free in every column: $\nabla_{\boldsymbol{z}} \cdot [\psi_k(\boldsymbol{z})\boldsymbol{\omega}] = 0$ for all $\boldsymbol{\omega} \in \mathbb{R}^3$.

Proof. Write $\boldsymbol{\xi} = \boldsymbol{M}_k{}^{-1}\boldsymbol{z}$ and, for an arbitrary reference field $\widehat{\boldsymbol{v}}$, let $\boldsymbol{v}(\boldsymbol{z}) = \det(\boldsymbol{M}_k)^{-1}\boldsymbol{M}_k\widehat{\boldsymbol{v}}(\boldsymbol{\xi})$. Since $\boldsymbol{M}_k$ is constant in $\boldsymbol{z}$, the chain rule gives, in index notation with implicit summation,

$$\begin{aligned} \nabla_{\boldsymbol{z}} \cdot \boldsymbol{v} = \frac{\partial v_i}{\partial z_i} &= \frac{1}{\det(\boldsymbol{M}_k)}\,(\boldsymbol{M}_k)_{ij}\,\frac{\partial \widehat{v}_j}{\partial \xi_\ell}\,(\boldsymbol{M}_k{}^{-1})_{\ell i} \\ &= \frac{1}{\det(\boldsymbol{M}_k)}\,\delta_{\ell j}\,\frac{\partial \widehat{v}_j}{\partial \xi_\ell} = \frac{1}{\det(\boldsymbol{M}_k)}\,\nabla_{\boldsymbol{\xi}} \cdot \widehat{\boldsymbol{v}}, \end{aligned} \tag{8}$$

using $(\boldsymbol{M}_k{}^{-1})_{\ell i}(\boldsymbol{M}_k)_{ij} = \delta_{\ell j}$. Taking $\widehat{\boldsymbol{v}}(\boldsymbol{\xi}) = \psi_{\mathrm{ref}}(\boldsymbol{\xi})\boldsymbol{\omega}$ – divergence-free by construction of ψ_{ref} – Eq. 8 gives $\nabla_{\boldsymbol{z}} \cdot [\psi_k(\boldsymbol{z})\boldsymbol{\omega}] = \det(\boldsymbol{M}_k)^{-1}\nabla_{\boldsymbol{\xi}} \cdot [\psi_{\mathrm{ref}}(\boldsymbol{\xi})\boldsymbol{\omega}] = 0$ for every $\boldsymbol{\omega}$, so every column of ψ_k is divergence-free. ■

Application to the anisotropic DFK. With $\boldsymbol{M}_k = \boldsymbol{R}_k\,\mathrm{diag}(\boldsymbol{a}_k)$ as in Eq. 4, the proposition applies to every kernel ψ_k, so any superposition

$$\boldsymbol{v}_\Theta(\boldsymbol{x}) = \bar{\boldsymbol{v}} + \sum_{k=1}^{K} \psi_k(\boldsymbol{x} - \boldsymbol{x}_k)\,\boldsymbol{\omega}_k \tag{9}$$

satisfies $\nabla_{\boldsymbol{x}} \cdot \boldsymbol{v}_\Theta(\boldsymbol{x}) = 0$ *exactly*, for arbitrary kernel weights, centers, rotations, and strictly positive semi-axis lengths $\boldsymbol{a}_k$. Because the map defined by $\boldsymbol{M}_k$ is linear, the same map also sends the reference kernel's unit-ball support to the ellipsoid $\{\boldsymbol{z} : \|\boldsymbol{M}_k{}^{-1}\boldsymbol{z}\| \le 1\}$: anisotropy reshapes the kernel's support without breaking compactness or exact divergence-freedom.

B EXPERIMENTAL DETAILS

We provide the full implementation details used in all experiments. Table 6 summarizes the DFK-Former architecture for both the static and temporal models, while Table 7 reports the corresponding training recipe, optimization settings, losses, and hardware configuration.

Table 6: DFK-Former architecture, as resolved for the reported static run. The upper block is the static model; the lower block gives the temporal model's additions and the two places where it departs from the static geometry. K is the kernel count and Δ the mean kernel spacing.

Component	Setting
Static model: backbone, heads and kernel geometry	
Token width d_{model}	256
Backbone blocks	6; each block = one masked cross-attention sublayer + one self-attention sublayer
Attention heads / head dimension	8 / 32
Feed-forward hidden width	$4\,d_{\text{model}} = 1024$, GELU
Dropout	0
Normalisation	pre-LayerNorm residual blocks; one final LayerNorm on the query set
Fourier position encoding γ	6 bands at frequencies $2^0 \dots 2^5$, raw coordinate concatenated (39 features)
Observation tokenizer	$\text{Linear}(43 \to 256) \to \text{GELU} \to \text{Linear}(256 \to 256)$
Tokenizer input dimension	39 (encoded position) + 3 (velocity) + 1 (confidence) = 43; 42 with the confidence channel disabled
Kernel-query tokens	K learnable 256-vectors plus a linear projection of $\gamma(\boldsymbol{x}_k^0)$
Register tokens	1
Prediction heads	$\text{Linear}(256 \to 256) \to \text{GELU} \to \text{Linear}(256 \to \cdot)$, all zero-initialised (weight and bias)
Head outputs	$\boldsymbol{\omega}_k$ (3); $\bar{\boldsymbol{v}}$ (3, from the register state); centre (3, tanh); radius (1, tanh); aspect (1, tanh); orientation (3, tanh)
Kernel count K	1024
Kernel centres	Poisson-disk (Mitchell best-candidate), seed 12345, fixed buffer
Kernel spacing	$\Delta = \text{box}/K^{1/3}$ on the canonical box of side 2
Radius mixture	scales $\{5.0, 4.0, 3.0\}\,\Delta$ at fractions $\{0.3, 0.5, 0.2\}$ of K (mean $\approx 4.1\,\Delta$)
Centre displacement bound	$\|\Delta \boldsymbol{x}_k\| \le 1.0\,\Delta$
Radius bound	$\sigma_k \in [\sigma_{\text{fixed}}/3,\ 3\sigma_{\text{fixed}}]$
Aspect bound	$a_{\text{long}}/a_\perp \in [\frac{1}{4}, 4]$, extend-only ($a_\perp = \sigma_k$)
Anisotropy construction	contravariant Piola transform (Eq. 4)
Parameters	10,228,750
Temporal model (Section 4.2)	
Backbone blocks	6; each block = spatial cross-attention + spatial self-attention + temporal self-attention
Time encoding	$[t,\ \sin(2\pi f t),\ \cos(2\pi f t)]_{f=1}^{4} \to \text{Linear}(9 \to 256)$, added to observation and query tokens
Temporal aggregation	per-frame kernel states mean-pooled over T before the Fourier heads
Fourier order F	2 (reported model); 7 for the ablation arm of Section E.1
Frames per example (training)	$T \sim \mathcal{U}\{6, \dots, 12\}$
Kernel geometry	one anisotropic kernel set shared across all frames: a single radius scale $3.5\,\Delta$ in place of the static model's mixture, fixed centres and radii, centre advection disabled. Only the Fourier weights and $\bar{\boldsymbol{v}}$ vary in t
Parameters	14,843,426; 14,976,038 with adaptive placement

Table 7: Training recipe of the reported static model, read from its fully resolved configuration. Ablation arms differ from it only in the single factor each ablation names (Section E); the temporal model's departures are in the last block. The auxiliary block acts on the placement and shape heads during training only and has no effect at inference. Per-stage data distributions are in Table 9; supervision point counts are in Table 8.

Setting	Value
Optimization and schedule	
Optimizer	AdamW, library-default β and ϵ; weight decay 0
Peak learning rate	3×10^{-4}
Schedule	constant to step $60{,}000$, cosine decay to step $145{,}000$, held at 10^{-5} thereafter; **no warm-up**
Gradient clipping	global norm 1.0
Batch size	32 fields per rank $\times$ 4 ranks $= 128$ fields per step
Steps	$150{,}000 =$ Core $60{,}000 +$ Hard $85{,}000 +$ cooldown $5{,}000$
Precision	model float32; bf16 autocast over the encoder forward only — kernel evaluation and all losses run in float32
Weight averaging	EMA, decay 0.9995; all reported evaluations use the EMA weights
Checkpoint selection	minimum of $\frac{1}{2}\left(\mathrm{EPE}_{\mathrm{GRF}} + \mathrm{EPE}_{\mathrm{CAF}}\right)$ on the frozen validation sets, scored on EMA weights, every 500 steps
Objective	
Objective	$\mathcal{L} = \mathcal{L}_{\mathrm{field}} + \lambda_{\mathrm{data}}\mathcal{L}_{\mathrm{data}} + \lambda_{\cos}\mathcal{L}_{\cos} + \lambda_{\mathrm{ortho}}\mathcal{L}_{\mathrm{ortho}} + \lambda_{\mathrm{cov}}\mathcal{L}_{\mathrm{cov}} + \lambda_{\mathrm{ctr}}\mathcal{L}_{\mathrm{ctr}} + \lambda_{\mathrm{rad}}\mathcal{L}_{\mathrm{rad}} + \lambda_{\mathrm{sh}}\mathcal{L}_{\mathrm{sh}}$; the last four are the auxiliary block below
λ_{field} / λ_{data}	1.0 / 1.0 (all stages)
$\lambda_{\cos}$	0.5 (all stages), linearly ramped over the first $3{,}000$ steps; magnitude floor 0.05
λ_{ortho}	0 (Core), 0.1 (Hard), 0.1 (cooldown)
Huber δ on the data term	1.0 (Hard and cooldown only)
Divergence	never in the objective; diagnostic only
Auxiliary regularisers on the placement and shape heads (training only)	
λ_{ctr} / λ_{rad}	0.01 / 0.01, all stages; keeps the predicted centres and radii near their fixed anchors
λ_{cov}	10^{-3} on Hard and cooldown; ramped over the first $20{,}000$ steps *of the Hard stage*, unramped thereafter. Soft repulsion against centre collapse
λ_{sh} (orientation warm-start)	0.1 annealed linearly to zero over $5{,}000$ steps, aspect factor 3.0; Core only. Target is computed from the input samples, never the clean field
Temporal model: departures from the above	
Batch size	16 fields per rank $\times$ 4 ranks $= 64$ fields per step
Objective	no auxiliary block: placement is fixed, so the coverage, centre, radius and shape terms are absent; $\mathcal{L}_{\mathrm{field}}$, $\mathcal{L}_{\mathrm{data}}$, $\mathcal{L}_{\cos}$ and $\mathcal{L}_{\mathrm{ortho}}$ carry the same weights and stage schedule
Checkpoint selection	validation EPE alone (no CAF term), EMA weights
Run provenance	
Seed	0; data seeds are rank-distinct
Hardware	$4\times$ NVIDIA A100 80GB PCIe
Wall clock	28 h 02 m (Core) + 31 h 20 m (Hard) + 2 h 20 m (cooldown), one uninterrupted launch

C Data Generation

This appendix specifies the data engine behind Section 4.3, summarized in Fig. 3. Every quoted value is the production setting used for the models of Section 5.

C.1 Conventions: domain, precision, and randomness

Fields are generated on the periodic unit cube ($L = 1$) using a fixed 64^3 FFT grid. Synthesis uses float64 precision, with float32 batches during training. Data generation is fully deterministic through seeds derived from the training step, sample index, and distributed worker identifier, making the

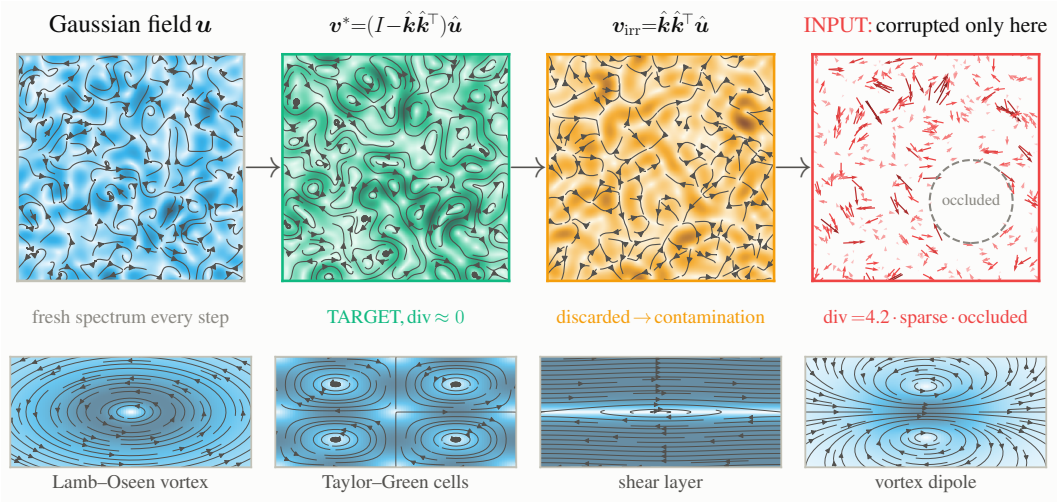

Figure 3: **Training-data pipeline** (2D shown, identical principles in 3D). *Top:* a Gaussian random field splits *exactly* in Fourier space into the solenoidal target $\boldsymbol{v}^*$ and an irrotational part, sparse observations are sampled and corrupted, and the resulting measurements are used to reconstruct the clean divergence-free target. Panel divergences are measured spectrally. *Bottom:* examples from the CAF library.

distribution independent of worker count. Training data are generated online, exposing the model to roughly 1.9×10^7 unique fields throughout training (4 workers $\times$ 32 fields $\times$ 150,000 steps).

C.2 Primary generator: band-limited Gaussian random fields with an exact Leray split

We shape three real white-noise grids w in Fourier space, $\widehat{\boldsymbol{u}} = A\,\widehat{w}$, using an anisotropic von-Kármán-like spectrum:

$$A(\boldsymbol{k}) = \frac{\sqrt{E(k_{\text{eff}})}}{\max(\|\boldsymbol{k}\|, 1)}, \qquad E(k) = \frac{(k/k_0)^4}{\big(1+(k/k_0)^2\big)^p}, \qquad k_{\text{eff}}^2 = \sum_{i=1}^{3}\Big(\frac{k_i}{a_i}\Big)^2. \tag{10}$$

Here $\boldsymbol{k}$ is the integer FFT index (cycles per domain), a_i are per-axis stretch factors, and the $\|\boldsymbol{k}\|$ divisor cancels the k^2 growth of modes per shell. As a result, E is the shell energy and k_0 is the energy-containing wavenumber. We zero the DC bin and remove modes with $\|\boldsymbol{k}\| > k_{\text{cut}}G$. The exact Leray projector then splits the field into the target and an irrotational contamination component:

$$\widehat{\boldsymbol{v}^*} = \big(I - \hat{\boldsymbol{k}}\hat{\boldsymbol{k}}^\top\big)\widehat{\boldsymbol{u}}, \qquad \widehat{\boldsymbol{v}}_{\text{irr}} = \hat{\boldsymbol{k}}\hat{\boldsymbol{k}}^\top\widehat{\boldsymbol{u}}, \qquad \hat{\boldsymbol{k}} = \boldsymbol{k}/\|\boldsymbol{k}\|. \tag{11}$$

Because this is a projection rather than a finite-difference subtraction, $\boldsymbol{v}^*$ has a spectral divergence RMS of $\sim 10^{-14}$ relative to the field RMS. Modes on the Nyquist hyperplanes (axis index $G/2$) have no conjugate partner, so the projector breaks Hermitian symmetry there. We zero these modes, which the band limit removes anyway.

Compactly supported divergence-free kernels integrate to zero, so the kernel sum in Eq. 3 is zero-mean and only $\bar{\boldsymbol{v}}$ carries net transport. To supervise this term, we add $\bar{\boldsymbol{v}} \sim \mathcal{N}\big(\mathbf{0}, (0.3\,\text{RMS}(\boldsymbol{v}^*))^2 I\big)$ after the split. The target remains divergence-free.

C.3 Capacity-matched band-limiting

We restrict the training spectra (Table 8) to scales that the output basis can represent. K kernels in the unit box have mean spacing $\Delta = L/K^{1/3}$ ($\Delta \approx 0.099$ at $K = 1024$). A compactly supported divergence-free basis cannot represent structure much finer than $\sim 2\Delta$, so supervision at finer scales adds an irreducible error floor. For example, a fixed basis reaches ≈ 0.9 relative EPE on broadband spectra, compared with ≈ 0.14 once the spectra are band-limited to the basis.

The spectral tail, rather than the peak alone, sets the effective length scale, so we constrain all three spectral parameters:

- $k_0 \in [1.5, 3]$ places the energy-containing wavelength at 3.4–6.7 kernel spacings.
- $p \in [3.5, 5]$ steepens the roll-off.
- $k_{\text{cut}} = 0.10$ sets the shortest retained wavelength at $\approx 1.6\Delta$, where almost no energy remains.

We also sample $a_i = 1 + \mathcal{U}[-0.3, 0.3]$ to prevent the model from learning an isotropy prior.

This design matches supervision to representational capacity. It does not assume that real flows are band-limited. Accordingly, the synthetic benchmark measures reconstruction within the basis's capacity, and the real-DNS benchmarks, which are not band-limited, are reported separately.

C.4 Coherent structures: the CAF library

Gaussian random fields lack the phase-coherent wakes, jets, shear layers and vortex tubes of real flows. The second generator superposes $n \sim \mathcal{U}\{1, \dots, 4\}$ closed-form structures from five families, each with a random center in $[0, L)^3$, random orientation, core radius $\sim \mathcal{U}[0.03, 0.15]\, L$ and strength $\sim \mathcal{U}[0.5, 2]$, and each divergence-free by a *different* argument; superposition inherits the property by linearity.

- **Lamb–Oseen vortex.** $\boldsymbol{v} = v_\theta(r)\, \hat{\boldsymbol{e}}_\theta$, $v_\theta(r) = \frac{\Gamma}{2\pi r}\big(1 - e^{-r^2/r_c^2}\big)$, r the cylindrical radius: any purely azimuthal field of r alone has $\nabla\cdot\boldsymbol{v} = r^{-1}\partial_\theta v_\theta = 0$. The singularity at $r \to 0$ is softened as $r_s = \sqrt{r^2 + \epsilon^2}$, a function of r alone, so the identity survives.
- **Taylor–Green cells.** $\boldsymbol{v} = A\big(\sin kx \cos ky \cos kz,\ -\cos kx \sin ky \cos kz,\ 0\big)$, integer k: the two nonzero divergence terms cancel term-by-term.
- **ABC / Beltrami flow.** $\boldsymbol{v} = (A \sin kz + C \cos ky,\ B \sin kx + A \cos kz,\ C \sin ky + B \cos kx)$: each component depends only on the *other two* coordinates, so every diagonal derivative vanishes.
- **Shear layer.** $\boldsymbol{v} = U \tanh\big((\boldsymbol{x} - \boldsymbol{x}_0)\cdot\boldsymbol{n}/\delta\big)\, \boldsymbol{e}$ with $\boldsymbol{e}$ Gram–Schmidt-orthogonalized against the layer normal $\boldsymbol{n}$ at build time, so $\nabla\cdot\boldsymbol{v} = f'(s)\,(\boldsymbol{n}\cdot\boldsymbol{e}) = 0$ exactly, for any profile.
- **Vortex ring.** A regularized Biot–Savart sum over 128 filament segments, $\frac{\Gamma}{4\pi}\sum_i \mathrm{d}\boldsymbol{l}_i \times \boldsymbol{r}_i/(\|\boldsymbol{r}_i\|^2 + a^2)^{3/2}$: each term is the exact curl of the smooth potential $\mathrm{d}\boldsymbol{l}_i/\sqrt{\|\boldsymbol{r}_i\|^2 + a^2}$, and $\nabla\cdot(\nabla\times\cdot) = 0$.

These families form the CAF benchmark and are also used to construct the per-family evaluation sets reported in the main paper.

C.5 Observation corruption

Corruption is applied only to the input; the continuous target is never modified.

Sampling geometries. For each field, we draw $M \sim \mathcal{U}\{M_{\text{lo}}, \dots, M_{\text{hi}}\}$ points in $[0, L)^3$ under periodic wrapping, using one of three geometries:

- *Uniform:* i.i.d. positions.
- *Clustered:* half the points fall in 3–12 Gaussian blobs of width $\mathcal{U}[0.03, 0.10]\, L$, and the rest are uniform. This mimics patchy Lagrangian measurements.
- *Surface:* a randomly oriented plane of thickness $0.02\, L$, mimicking a camera-visible sheet.

Velocity corruption. At the sampled positions,

$$\boldsymbol{u}_m = \boldsymbol{v}^*(\boldsymbol{p}_m) + \chi\, \boldsymbol{v}_{\text{irr}}(\boldsymbol{p}_m) + \sigma\, \boldsymbol{\epsilon}_m, \qquad \boldsymbol{\epsilon}_m \sim \mathcal{N}(\boldsymbol{0}, I_3), \qquad \sigma = \nu\, \mathrm{RMS}(\boldsymbol{v}^*), \tag{12}$$

with $\chi \sim \mathcal{U}[0, \chi_{\max}]$ and $\nu \sim \mathcal{U}[\nu_{\text{lo}}, \nu_{\text{hi}}]$ drawn per field. For spectral fields, $\boldsymbol{v}_{\text{irr}}$ is the target's own Leray complement (Eq. 11), so the divergent contamination is physically consistent with the target

rather than arbitrary noise. The analytic generators have no such complement. For them, $\boldsymbol{v}_{\mathrm{irr}} = \nabla\phi$ for a random sum of 8 low-frequency sinusoids, rescaled to the target RMS.

Occlusion. With probability p_{occ}, we delete all samples inside a ball occupying a volume fraction $\mathcal{U}[0.05, 0.35]$. Distances use the minimum-image convention, so the ball wraps correctly across the periodic boundary.

C.6 Normalization and supervision

Per-field normalization. We normalize each example before tokenization and invert the transform on output. Positions are mapped to a canonical cube, $\boldsymbol{p} \mapsto (\boldsymbol{p} - \frac{L}{2}\mathbf{1})\,(B/L)$ with $B = 2$. Velocities are rescaled by $s = 1/\mathrm{RMS}(\boldsymbol{u})$. We compute s from the observed, corrupted input rather than from the target, so the same transform is available at deployment. The target remains in world coordinates: canonical query points are mapped back, the target is evaluated there, and the result is scaled by the same s. The transform is therefore exactly invertible.

Supervision. During training, the field loss is evaluated at 2048 Monte-Carlo query points per example, resampled at every step, so no location is supervised twice. Evaluation instead uses 4096 fixed query points per field. Batches are zero-padded to $M_{\max}$ observations. A binary mask excludes padded entries from attention and the losses, and the data loss is normalized per example. Within each curriculum stage, the generator for each example is drawn independently according to that stage's mixture weights (Table 9).

Table 8: Data-generation parameters. Ranges are drawn independently per field; $\mathcal{U}$ denotes a uniform draw (continuous, or over integers where the range is written $\{\cdot\}$). Corruption entries give the values outside the curriculum; per-stage overrides are in Table 9.

Symbol	Meaning	Value / range
Domain and precision		
L	periodic domain size, $[0, L)^3$	1.0
G	FFT generation grid	64^3
—	generation / training precision	float64 / float32
Spectral generator (Eq. 10)		
k_0	energy-containing wavenumber (cycles/domain)	$\mathcal{U}[1.5, 3.0]$
p	spectral roll-off exponent	$\mathcal{U}[3.5, 5.0]$
a_i	per-axis anisotropy stretch	$1 + \mathcal{U}[-0.3, 0.3]$
k_{cut}	hard cutoff, $\|\boldsymbol{k}\| > k_{\mathrm{cut}}G$	0.10
$\bar{\boldsymbol{v}}$	injected constant mean flow	$\mathcal{N}(\mathbf{0}, (0.3\,\mathrm{RMS})^2 I)$
Coherent-structure library (CAF)		
n	superposed structures per field	$\mathcal{U}\{1, \ldots, 4\}$
r_c	core / tube radius	$\mathcal{U}[0.03, 0.15]\,L$
—	strength (Γ, amplitude, or speed)	$\mathcal{U}[0.5, 2.0]$
—	ring loop-to-core ratio	$\mathcal{U}[3, 8]$
—	Taylor–Green / ABC integer wavenumber	$\mathcal{U}\{1, 2, 3\}$
Corruption (cf. Eq. 12)		
M	input samples per field	$\mathcal{U}\{256, \ldots, 4096\}$
—	sampling geometry	uniform / clustered / surface
χ	irrotational contamination	$\mathcal{U}[0, \chi_{\max}]$
ν	noise, relative to target RMS	$\mathcal{U}[0, \nu_{\max}]$
p_{occ}	occlusion probability	0.5
—	occluded volume fraction	$\mathcal{U}[0.05, 0.35]$
Supervision		
—	MC query points per example (train)	2048
—	frozen query points per field (eval)	4096
—	fields per step (4 ranks $\times$ 32)	128

C.7 CURRICULUM

Three stages change only the data distribution and the loss weights; the model is untouched (cf. Table 9). Core learns the projection itself under mild corruption; Hard, with heavier contamination and noise, a smaller sample budget, occlusion, and coherent-structure mixing, is where robustness to heavy corruption and coherent structure is acquired; a short cooldown de-specializes the final checkpoint from the hardest regime.

Table 9: Training curriculum. Only the data distribution and loss weights change between stages. "–" means the term is off. $\chi_{\max}$ is the maximum irrotational contamination, $\nu_{\max}$ the maximum relative noise, p_{occ} the occlusion probability.

Stage	Steps	Sources (spectral / coherent)	$\chi_{\max}$	$\nu_{\max}$	M	p_{occ}
Core	60,000	1.0 / –	0.5	0.05	$[512, 4096]$	–
Hard	85,000	0.6 / 0.4	1.5	0.20	$[128, 2048]$	0.5
Cooldown	5,000	0.7 / 0.3	1.0	0.15	$[256, 4096]$	0.5

C.8 TIME-VARYING DATA

Temporal targets are divergence-free at every instant by construction, not by finite differencing. We use two families.

Modal-spectral (deformation). We Leray-split $n_{\text{modes}} \sim \mathcal{U}\{2, 3, 4\}$ spectral fields $\boldsymbol{v}_j$ as in Section C.2 and combine them as

$$\boldsymbol{v}(\boldsymbol{x}, t) = \bar{v} + \sum_j c_j(t)\, \boldsymbol{v}_j(\boldsymbol{x}), \qquad t \in [0, 1], \tag{13}$$

where each c_j is a real Fourier series in t. The weights do not depend on space, so the sum is divergence-free at every t.

Advected coherent structures (transport). We rigidly translate structures from Section C.4 at constant velocity,

$$\boldsymbol{v}(\boldsymbol{x}, t) = \sum_i \boldsymbol{s}_i(\boldsymbol{x} - \boldsymbol{a}_i t), \tag{14}$$

where each $\boldsymbol{a}_i$ has a random direction and $\|\boldsymbol{a}_i\| \sim \mathcal{U}[0, 0.3]$ domain lengths per unit time. Divergence commutes with spatial shifts, so this construction is exactly divergence-free. We avoid nonlinear self-advection, which would require a pressure projection.

Temporal band-limiting. Following Section C.3, we match the temporal content of the data to the model's capacity. An order-F temporal expansion represents only harmonics up to F, so we use $F_{\text{data}} = 9$ harmonics with amplitudes decaying as $(1 + f)^{-1}$.

Frames, corruption, and supervision. Each step draws $T \sim \mathcal{U}\{6, \ldots, 12\}$ frames at times $t_i = i/(T - 1)$. We corrupt each frame independently (Section C.5), which forces the model to fuse information across time. A single normalization pooled over all frames keeps the sequence consistent. Query times for the field loss are drawn from $\mathcal{U}[0, 1]$ rather than at the observed frame times, so the model is also trained to reconstruct at unobserved times.

C.9 FROZEN EVALUATION SETS

Each evaluation set is generated once from a fixed base seed, disjoint from training, with per-field seed = base + index. Sets are stored as HDF5 files containing the normalized inputs, frozen query points, targets, and per-field provenance, so every reported number traces back to an exact field. Synthetic sets use the same band-limited configuration as training (Section C.3). The real-DNS sets are not band-limited and are reported separately. Table 10 gives the composition of the 582-flow benchmark.

Table 10: Composition of the 582-flow evaluation benchmark. Synthetic sets use seeds disjoint from training. The CAF and single-family sets draw on the training structure library (Table 9), so they measure in-distribution accuracy; held-out families are evaluated in Table 21. JHTDB sets are real DNS (Li et al., 2008), resampled to 64^3 and used for evaluation only.

Set	Generator	Fields	Role
in-dist	spectral, training band-limit	128	matched-distribution accuracy
CAF	coherent-structure mixture	64	coherent structure (in training)
Taylor–Green	single analytic family	64	per-family accuracy
Lamb–Oseen	single analytic family	64	per-family accuracy
vortex ring	single analytic family	64	per-family accuracy
shear layer	single analytic family	64	per-family accuracy
ABC	single analytic family	64	per-family accuracy
isotropic turbulence	JHTDB (real DNS)	18	zero-shot real DNS
forced MHD turbulence	JHTDB (real DNS)	18	zero-shot real DNS
channel flow	JHTDB (real DNS)	17	zero-shot, wall-bounded
transitional b. layer	JHTDB (real DNS)	17	zero-shot, non-periodic
Total		**582**	

C.10 Verification

Grid-based targets. For Leray-split targets, a spectral divergence operator, which is exact for band-limited periodic fields, gives a divergence RMS of $\sim 10^{-14}$ relative to the field RMS.

Analytic targets. These targets have no grid, so we verify them by automatic differentiation of the continuous field. Relative residuals are $\lesssim 10^{-14}$ for all five families and their superpositions. This holds both at random query points and near vortex-ring filaments, where the Biot–Savart sum is most strongly cancelling.

Temporal targets and inputs. Temporal targets pass the same checks at multiple times per sequence. Input divergence is $O(\chi)$ and clearly nonzero (Fig. 3), so the reconstruction task is not trivial. Unit tests additionally cover the normalization round-trip and bit-exact reproducibility across processes.

D From the Original DFK Representation to Ours: A Fitting Study

We analyze the proposed modifications to the divergence-free kernel (DFK) representation of Ni et al. (2025) using *fitting experiments only*. Least-squares fits are performed on clean divergence-free fields without learning, corruption, or sparse observations, making the results properties of the representation itself. The study evaluates four modifications: (i) an explicit mean-flow term, (ii) spacing-relative support radii with strong kernel overlap, (iii) a multi-scale radius mixture, and (iv) anisotropic kernels.

D.1 Protocol

All experiments use a common protocol. Targets comprise band-limited spectral flows (Gaussian Random fields), analytic divergence-free fields, and DNS snapshots from JHTDB (Li et al., 2008). Coefficients are fitted by ridge least squares, reporting the best ridge value, and evaluated on query points disjoint from those used for fitting. Because the representation is linear in $\{\boldsymbol{\omega}_k\}$ and $\bar{\boldsymbol{v}}$ for fixed centers, radii, and shapes, the resulting solution is the optimal approximation within the selected basis. All methods share identical Poisson-disk center placement (Bridson, 2007), use float64 precision, and enforce divergence-freeness by construction.

D.2 A constant mean-flow term

Every compactly supported kernel integrates to zero: $\int(-\mathbf{I}\nabla^2 + \nabla\nabla^\top)\phi \,\mathrm{d}\boldsymbol{z} = 0$, implying that a finite kernel sum cannot represent a uniform flow component. Since the Leray projector preserves the mean, any target with non-zero transport lies partly outside the original span. Adding a constant velocity term $\bar{\boldsymbol{v}}$ resolves this limitation.

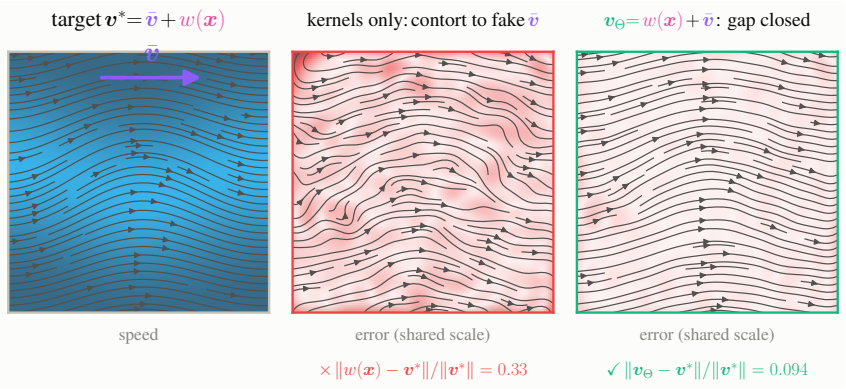

Figure 4: **The mean-flow gap.** Kernels alone cannot represent net transport because their sum is zero-mean. Adding a mean-flow term substantially reduces approximation error.

Figure 4 illustrates the effect on a flow composed of a uniform component and a zero-mean vortical field. Using kernels alone yields relEPE 0.334, whereas adding $\bar{\boldsymbol{v}}$ reduces the error to 0.094 while accurately recovering the underlying mean velocity. The term is therefore essential for flows with non-zero mean and incurs no cost on zero-mean fields.

D.3 Support radius: the kernels must overlap heavily

The packing-based radius of Ni et al. (2025) yields supports smaller than the mean kernel spacing $\Delta = (V/K)^{1/3}$, resulting in limited overlap between neighboring kernels. For divergence-free kernels, this regime poorly captures smooth large-scale structures, whereas approximation quality improves substantially as overlap increases. We therefore parameterize the support radius as

$$\sigma_k = \text{scale} \cdot \Delta,$$

and tune the radius scale directly rather than the packing coefficient. Table 11 evaluates the best achievable approximation on real DNS fields at fixed $K = 8192$. Increasing overlap consistently lowers the approximation floor across all datasets, often by more than an order of magnitude, indicating that support overlap is a more important design factor than kernel count.

Table 11: Best achievable fit (relEPE) of the anisotropic DFK span to clean real-DNS fields (Li et al., 2008) at fixed $K = 8192$, as a function of the support radius expressed in kernel spacings. Overlap is the dominant design variable: the floor falls monotonically as the kernels overlap more.

Radius / spacing	channel	transitional b.l.	isotropic turb.	forced MHD
1.5	0.106	0.165	0.483	0.739
2.5	0.037	0.056	0.173	0.261
3.5	0.012	0.019	0.050	0.075
4.5	0.010	0.016	0.040	0.058
6.0	**0.009**	**0.014**	**0.035**	**0.051**

The opposite failure: over-redundancy. Although larger radii consistently improve approximation quality, excessive overlap can make neighboring kernels highly redundant, reducing the effective rank of the basis and degrading numerical conditioning. In our proposed model, we employ a multi-scale radius mixture with scales $\{5.0, 4.0, 3.0\}$ and proportions $\{0.3, 0.5, 0.2\}$. This choice maintains the strong overlap required to represent smooth large-scale flow structures, while distributing kernels across several support sizes to improve coverage of multiple spatial scales and limit redundancy.

D.4 Anisotropic kernels

The original DFK basis is isotropic, whereas many flow structures such as vortex tubes, shear layers, and jets are elongated. Using the Piola-transformed anisotropic kernel from Eq. 4 preserves exact

Table 12: Anisotropic vs isotropic fit (relEPE) on a vortex-tube target: identical centers, radius scale $3.5\,\Delta$, Piola construction, aspect ratio 4, 8000 fit samples, 4000 scoring points, ridge 10^{-4}. Same-memory reduces the anisotropic count to $\lfloor 7K/11 \rfloor$. The equal-volume column shows that preserving support volume is worse than not using anisotropy at all.

		same-K		same-memory
K	isotropic	aniso, equal-volume	aniso, extend-only	aniso, extend-only
256	0.173	0.251	**0.139**	0.173
512	0.137	0.247	**0.092**	**0.122**
1024	0.128	0.282	**0.055**	**0.076**
2048	0.124	0.356	**0.038**	**0.045**
4096	0.215	0.592	**0.049**	**0.038**

divergence-freeness while adapting support to directional structures. Table 12 compares isotropic and anisotropic kernels on a vortex-tube target using identical center placement, radius scale, and ridge regularization, with orientations derived from local vorticity. Anisotropic kernels consistently improve accuracy at equal kernel count and remain advantageous under equal-memory comparisons, despite requiring additional parameters per kernel. Three conclusions emerge. First, preserving the transverse radius while elongating the kernel ("extend-only": $(a_\perp = \sigma_k, a_{\text{long}} = \sigma_k \rho)$) is consistently superior to equal-volume scaling ($a_\perp = \sigma_k/\rho^{1/3}$, $a_{\text{long}} = \sigma_k \rho^{2/3}$), which can even underperform the isotropic basis. Second, orientation alignment helps only when coherent directional structure is present; on isotropic spectral flows, aligned and random orientations perform similarly. Third, cylindrical anisotropy provides the best accuracy-memory trade-off, outperforming more expressive triaxial parameterizations once parameter cost is accounted for.

D.5 What fitting alone does *not* justify: adaptive placement

Placement is the primary source of remaining approximation error in the fitting study, motivating the adaptive strategy of Section 4.1. An oracle experiment that optimizes kernel centers and radii directly against the target field yields mixed results: improvements appear on flows requiring precise localization (isotropic turbulence $0.107 \to 0.097$, forced MHD $0.161 \to 0.126$), while performance can deteriorate when the fixed basis already provides a good representation (channel $0.025 \to 0.043$, transitional boundary layer $0.043 \to 0.056$), highlighting the difficulty of the underlying non-convex optimization. Furthermore, adaptive placement increases the parameter count from three kernel weights to seven parameters per kernel, making reliable estimation substantially more data demanding. Together, these results suggest that adaptivity is valuable, but should be learned from data rather than optimized independently for each field.

D.6 Kernel normalization with per-kernel radii

When kernel radii vary, the DFK operator acquires an additional factor of ${\sigma_k}^{-2}$ from the chain rule applied to the normalized coordinate $r = \|\boldsymbol{z}\|/\sigma_k$. Divergence-freeness is preserved regardless of this factor, but omitting it alters the relative amplitudes of kernels with different radii. While inconsequential when a single radius is used throughout, correct normalization becomes essential in our setting because the representation combines multiple radius scales (Section D.3) and predicts radii adaptively for each example. We therefore include the ${\sigma_k}^{-2}$ factor so that kernels with different support sizes contribute at consistent physical scales, ensuring comparable parameterizations across both multi-scale and adaptively predicted radii.

E Additional Ablation Studies

This appendix provides additional ablation studies that are absent from the main paper.

E.1 Temporal Fourier order

We study the effect of the temporal Fourier order F used to parameterize the kernel coefficients (Section 4.2). The comparison varies only the value of F (2 vs. 7), keeping kernel geometry, training data, and optimization settings fixed. On the 60-flow real-DNS interpolation benchmark, the lower-order model (F=2) achieves lower median EPE (0.114 vs. 0.128) and mean angular error ($4.8°$ vs. $5.7°$). The largest gains occur on wall-bounded flows. This result should be interpreted as a capacity-matching effect rather than evidence that smaller F is universally preferable. On the synthetic validation distribution, which contains stronger high-frequency temporal content, the higher-order model (F=7) performs better. As in Section C.3, matching model capacity to the temporal complexity of the data is more important than maximizing expressiveness.

E.2 Controlled Architecture Matrix

Table 13 summarizes single-factor ablations on a shared reduced-scale baseline. The results identify the set encoder and kernel representation as the dominant contributors to performance. Replacing attention with DeepSets pooling substantially degrades accuracy, while increasing the number of kernels and enabling adaptive placement both provide clear gains. In contrast, architectural scaling and optimization details have smaller effects. Width and Fourier positional encoding yield modest but consistent improvements when evaluated with a sufficiently long training budget, whereas depth and the orthogonality-loss weight have only minor impact ($\leq 1.5\%$). Following these observations, small effects are reported using extended training schedules, while short-budget measurements are interpreted only as directional trends. The kernel-count result (K: $512 \to 2048$) should likewise be viewed as a directional signal; the full-resolution sweep in Section 5.4 provides a more precise characterization of the accuracy–capacity trade-off.

Table 13: Controlled architecture matrix: single-factor ablations on a shared reduced-scale base (K=512, d_{model}=128, 4 blocks, Core stage only), scored on the 128-field Gaussian Random Fields set. These are ablation *signal* runs, not paper-scale numbers; small effects are quoted at the $10\times$ step budget (Section E.2). Negative Δ is an improvement.

Factor	Change	Δ EPE	Budget
Set encoder	attention $\to$ DeepSets pooling	$+49.1\%$	3k steps
Kernel count	K: $512 \to 2048$ (no 1024 point)	-12.0%	3k steps
Adaptive placement	fixed $\to$ predicted centers/radii	-7.4%	3k steps
Width	d_{model}: $128 \to 384$	-4.8%	30k steps
Fourier position enc.	off $\to$ on	-2.4%	30k steps
Orthogonality loss	λ_{ortho}: $0.1 \to 0$	$+0.04\%$	30k steps

E.3 Single-axis corruption sweep

We evaluate robustness by varying one corruption factor at a time while keeping the others fixed: sample count, noise level, occluded fraction, and target spectral content. Figure 5 compares multiple baselines across the full range of operating conditions, while Tables 14 and 15 summarize the hardest settings and a higher-statistics comparison against DFK fitting. Across all operating points, DFK-Former is the strongest feed-forward method that exactly preserves incompressibility. Gains are particularly pronounced over optimization-based DFK fitting with adaptive kernels, while FNO and DeepONet degrade more noticeably under sparse observations, strong corruption, and out-of-distribution spectra. A direct-regression ablation built on the same encoder attains similar accuracy but exhibits substantial divergence leakage, highlighting the role of the DFK representation. The comparison against fixed-kernel DFK fitting reveals a consistent advantage across all corruption axes. The margin narrows under severe corruption, where optimization-based fitting remains relatively robust, but remains largely stable as the target spectrum moves beyond the training range.

F Additional Results

In this section, we will present additional results and comparisons to evaluate DFK-Former against the different defined baselines.

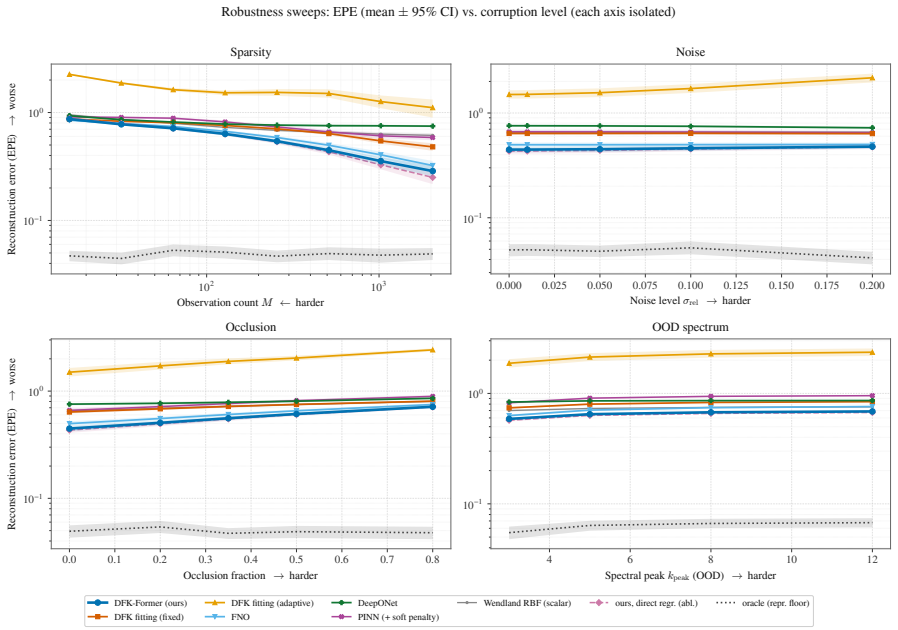

Figure 5: **Axis-isolated robustness sweeps** Reconstruction error (EPE; mean ± 95% bootstrap CI; 72 fields per point) as a function of corruption strength along a single axis, with all other factors fixed to nominal settings. DFK-Former (blue) is the strongest, exactly divergence-free feed-forward method. The dashed pink curve is an ablated variant of DFK-Former, while the grey dotted line denotes the oracle representation floor obtained by least-squares fitting the basis to the clean target, i.e., the best achievable error within the representation.

Table 14: Multi-method sweep at each axis's hard endpoint (M=64; σ_{rel}=0.2 at M=512; occluded fraction 0.8; target $k_{\text{peak}} \in [12, 18]$). EPE, lower is better; the last column is residual divergence. The row before the floor is our own ablation, not a baseline: on the identical backbone it matches our accuracy and leaks divergence.

Method	M=64	noise 0.2	occl. 0.8	$k_{\text{peak}}[12, 18]$	$\|\nabla\cdot v\|_2$
DFK-Former (ours)	**0.714**	0.477	**0.713**	0.687	2–7 $\times 10^{-7}$
FNO	0.742	0.501	0.749	0.760	0.40–0.67
DeepONet	0.816	0.723	0.856	0.862	≈ 0.2
DFK fitting (fixed kernels)‡	0.804	0.636	0.805	0.842	$\sim 10^{-16}$
DFK fitting (adaptive kernels)	1.628	2.167	2.425	2.361	$\sim 10^{-15}$
Wendland RBF (scalar)	0.795	0.657	0.803	0.756	1.5–3.1
PINN (+ soft penalty)	0.886	0.649	0.894	0.954	0.83–1.5
Ablation of our own model, not a baseline					
Set-Transf. + MLP (direct) †	0.715	**0.475**	**0.713**	**0.672**	0.24–0.76
oracle representation floor	*0.053*	*0.041*	*0.048*	*0.068*	$\sim 10^{-15}$

† Identical encoder, value decoder in place of the DFK head (Section 5.4). ‡ At its per-point best-EPE ridge.

F.1 Per-set breakdown of the main benchmark

Table 16 reports per-set results for the main benchmark. DFK-Former achieves the lowest EPE on all eleven evaluation sets, with no competing method outperforming it on any column. Interestingly, the real DNS benchmarks are not the most challenging cases: all methods obtain lower errors on the four JHTDB subsets than on the Gaussian Random Fields (GRF) set. For DFK-Former, the average EPE drops from 0.343 on GRF to 0.153 across the JHTDB subsets. As discussed in Section C.3, GRF concentrates energy near the representation's spectral limit, making it a particularly demanding test of approximation capacity. Finally, the two curl-of-potential INRs perform worse than the trivial

Table 15: Single-axis corruption sweep: model EPE against per-field DFK fitting (fixed kernels) at its per-point best ridge, 500 fields $\times$ 3 seeds per point, 95% bootstrap confidence intervals non-overlapping at every point. The margin is $1 - \mathrm{EPE}_{\mathrm{model}}/\mathrm{EPE}_{\mathrm{fit}}$; the last column repeats it at the hard endpoint against a fit over the model's own anisotropic basis.

Axis	Range (easy $\to$ hard)	model EPE	margin vs fit	margin vs aniso. fit
Sample count M	$2048 \to 64$	$0.315 \to 0.713$	$36.3\% \to 11.2\%$	$\to 15.8\%$
Noise ν	$0 \to 0.4$	$0.326 \to 0.390$	$36.8\% \to 26.5\%$	$\to 27.7\%$
Occluded fraction	$0 \to 0.5$	$0.306 \to 0.492$	$36.2\% \to 26.0\%$	$\to 29.5\%$
Target $k_{\min}$–$k_{\max}$	$[3,5] \to [12,18]$	$0.441 \to 0.544$	$29.3\% \to 32.0\%$	$\to 38.0\%$

zero-field predictor (relative $L^2 > 1$) on every synthetic set, indicating that exact incompressibility alone does not provide a sufficiently strong inductive bias for accurate reconstruction.

Table 16: **Per-set breakdown of the main benchmark** (EPE, lower is better). **Bold** marks the column minimum. *GRF* is the frozen Gaussian Random Fields set and *CAF* the coherent-structure mixture (Table 10). No method has trained on JHTDB, so those four columns are zero-shot throughout.

	Synthetic (band-limited)							Real DNS (zero-shot)			
Method	GRF	CAF	Taylor–Gr.	Lamb–Os.	vtx. ring	shear	ABC	iso.	MHD	chan.	tr. BL
Wendland RBF (scalar)	0.634	0.615	0.607	0.627	0.560	0.622	0.618	0.566	0.553	0.601	0.572
SIREN (+ soft penalty)	0.550	0.537	0.560	0.435	0.444	0.438	0.418	0.256	0.310	0.183	0.180
Curl-SIREN	0.998	1.177	1.167	1.102	1.028	1.108	1.005	0.673	0.729	0.769	0.699
Vector-potential INR	0.819	1.029	0.969	0.943	0.942	0.989	0.821	0.469	0.470	0.623	0.485
DFK fitting (fixed kernels)	0.545	0.643	0.482	0.715	0.692	0.743	0.548	0.752	0.732	0.889	0.842
DFK fitting (adaptive kernels)	0.602	0.603	0.536	0.520	0.462	0.575	0.527	0.546	0.566	0.575	0.488
FNO	0.375	0.254	0.146	0.263	0.281	0.333	0.136	0.251	0.311	0.166	0.167
FNO (+ Leray projection)	0.372	0.270	0.139	0.262	0.296	0.420	0.129	0.246	0.305	0.163	0.162
DeepONet	0.745	0.443	0.682	0.324	0.411	0.399	0.280	0.371	0.454	0.229	0.248
DeepONet (+ Leray projection)	0.744	0.456	0.681	0.347	0.429	0.486	0.278	0.366	0.451	0.223	0.242
PINN (+ soft penalty)	0.566	0.449	0.489	0.335	0.330	0.325	0.325	0.228	0.294	0.074	0.092
HDNet-3D	0.391	0.196	0.143	0.146	0.188	0.165	0.105	0.266	0.301	0.201	0.216
DFK-Former (ours)	**0.343**	**0.101**	**0.053**	**0.056**	**0.062**	**0.054**	**0.036**	**0.200**	**0.269**	**0.063**	**0.081**

F.2 Enforcing Exact Divergence-Freeness in Operator-Learning Baselines

A natural question is whether the advantage of DFK-Former stems primarily from exact incompressibility. To isolate this factor, we post-process the predictions of the operator-learning baselines (FNO and DeepONet), whose raw outputs exhibit non-zero divergence (0.71 and 0.23, respectively), using a standard Leray projection. All experiments are conducted on the same eleven evaluation sets with identical metrics and aggregation procedures.

Leray projection. For each method, the predicted velocity field $\boldsymbol{u}$ is represented on a regular $[3, G, G, G]$ grid over the canonical domain, transformed to Fourier space, projected onto the divergence-free subspace, and evaluated at arbitrary query locations via a spectral read-out:

$$\widehat{\boldsymbol{v}}_{\mathrm{sol}}(\boldsymbol{k}) = \left(I - \hat{\boldsymbol{k}}\hat{\boldsymbol{k}}^{\top}\right)\widehat{\boldsymbol{u}}(\boldsymbol{k}), \qquad \boldsymbol{v}_{\mathrm{sol}}(\boldsymbol{x}) = \frac{1}{G^3}\sum_{\boldsymbol{k}} \widehat{\boldsymbol{v}}_{\mathrm{sol}}(\boldsymbol{k}) e^{\,i\,\boldsymbol{k}\cdot\boldsymbol{x}}, \qquad \hat{\boldsymbol{k}} = \boldsymbol{k}/\|\boldsymbol{k}\|, \tag{15}$$

using the same angular wavenumbers $2\pi\boldsymbol{k}/L$ as Eq. 11. Since $\boldsymbol{k}\cdot\widehat{\boldsymbol{v}}_{\mathrm{sol}}(\boldsymbol{k}) = 0$ for every retained Fourier mode,

$$\nabla\cdot\boldsymbol{v}_{\mathrm{sol}}(\boldsymbol{x}) = \frac{1}{G^3}\sum_{\boldsymbol{k}} i\,(\boldsymbol{k}\cdot\widehat{\boldsymbol{v}}_{\mathrm{sol}}(\boldsymbol{k}))e^{\,i\,\boldsymbol{k}\cdot\boldsymbol{x}} \equiv 0,$$

yielding a field that is divergence-free everywhere, not only at grid locations. This property requires spectral evaluation; replacing it with trilinear interpolation breaks the commutativity between projection and reconstruction and increases the divergence norm to 0.23 for FNO and 0.048 for DeepONet. FNO is projected directly from its native grid, whereas DeepONet is first evaluated on a G^3 grid before projection. The mean-flow component is preserved ($\widehat{P}(\mathbf{0}) = I$), Nyquist hyperplanes are removed as in Eq. 11, and all computations are performed in `float64`.

Exact incompressibility does not recover accuracy. The projection incurs only a modest runtime overhead ($1.15\times$ for FNO and $2.15\times$ for DeepONet) and changes macro EPE by $+2.9\%$ and $+2.5\%$, respectively. Despite becoming exactly divergence-free, the projected models remain significantly behind DFK-Former, by factors of $2.10\times$ (FNO) and $3.57\times$ (DeepONet) in macro EPE.

Projection helps most datasets. The aggregate degradation is largely driven by a single failure mode. Across the eleven evaluation sets (Table 16), projection improves EPE on $8/11$ datasets for FNO and $7/11$ for DeepONet, including all four real-DNS datasets. The macro-level penalty originates almost entirely from the shear-layer case, where FNO degrades from 0.333 to 0.420 and DeepONet from 0.399 to 0.486. Excluding this dataset, projection changes macro EPE by only -0.3% for FNO and $+0.7\%$ for DeepONet. Ablating the pipeline shows that the degradation arises from the projection itself rather than the spectral read-out. For FNO on the shear layer, spectral evaluation alone increases EPE by only 1.0% ($0.333 \to 0.336$), while the remaining degradation is introduced by the projection. This behavior reflects the periodic-domain assumption of FFT-based solenoidal projection: the shear-layer example contains a strong discontinuity across the periodic boundary, which is interpreted as an irrotational component and consequently removed. In contrast, **DFK-Former is mesh-free and imposes no periodic-boundary assumption**.

Grid resolution is not the limiting factor. Since DeepONet requires discretization before projection, we additionally varied the projection grid resolution. Increasing the grid from $G = 32$ to $G = 64$ changes macro EPE only from 0.4275 to 0.4253 (0.5% improvement) despite an eightfold increase in spectral degrees of freedom. We therefore report results for $G = 32$, which also matches FNO's native resolution.

F.3 Physical consistency: divergence, energy, enstrophy and spectrum

Table 17 reveals a clear separation between three classes of methods. Methods that enforce incompressibility *by construction* achieve divergence at numerical precision, including curl-based INRs, DFK fitting, Leray-projected operators, and DFK-Former. Soft-constrained methods reduce divergence but do not eliminate it, as the penalty is enforced only at sampled locations. Unconstrained methods exhibit divergence errors comparable to the scale of the reconstructed field. This separation remains unchanged across all ablations (Section E).

Table 17: **Divergence quality**, macro-averaged over the eleven static sets; the L_2 column is in Table 1. Our $\sim 10^{-7}$ is the `float32` autograd diagnostic's floor, not a property of the field: the DFK construction is solenoidal for any coefficients, centers and radii (Section A).

Method	$\|\nabla\cdot\boldsymbol{v}_\Theta\|_\infty \downarrow$	div-free
Wendland RBF (scalar)	1.9×10^{1}	no
SIREN (+ soft penalty)	7.9×10^{0}	no (soft)
Curl-SIREN	5.4×10^{-14}	by construction
Vector-potential INR	2.7×10^{-14}	by construction
DFK fitting (fixed kernels)	2.8×10^{-14}	by construction
DFK fitting (adaptive kernels)	4.3×10^{-14}	by construction
FNO	3.5×10^{0}	no
FNO (+ Leray projection)	5.0×10^{-16}	by construction
DeepONet	8.9×10^{-1}	no
DeepONet (+ Leray projection)	2.5×10^{-16}	by construction
PINN (+ soft penalty)	2.8×10^{0}	no (soft)
HDNet-3D	4.1×10^{0}	partially
DFK-Former (ours)	2.6×10^{-6}	by construction

Beyond divergence, Table 18 evaluates kinetic energy (KE), enstrophy (Ω), and energy-spectrum fidelity. DFK-Former achieves the lowest band-integrated spectrum error on both the turbulence benchmark and the broader ten-family evaluation, indicating the most faithful recovery of multiscale dynamics. While optimization-based methods with soft divergence penalties can achieve slightly lower KE or enstrophy errors on some turbulence datasets, they do so while violating incompressibility and requiring orders of magnitude higher inference cost. Conversely, exact incompressibility alone

Table 18: **Physical-consistency battery**, macro-averaged over the three turbulence sets (Gaussian Random Fields, JHTDB isotropic, JHTDB MHD; 128+18+18 fields; spectral curl on the node-aligned 64^3 grid). The last two columns repeat enstrophy and $E(k)$ over the ten-family battery, available for the three methods run over all families.

	Three turbulence sets			Ten-family battery	
Method	rel. KE ↓	rel. Ω ↓	band $E(k)$ ↓	rel. Ω ↓	band $E(k)$ ↓
SIREN (+ soft penalty)	**0.077**	**0.296**	0.175	–	–
Curl-SIREN	1.388	4.962	1.576	–	–
DFK fitting (fixed kernels)	0.148	0.486	0.230	–	–
FNO	0.202	0.696	0.258	2.660	0.211
DeepONet	0.388	0.875	0.435	–	–
PINN (+ soft penalty)	0.104	0.430	0.208	–	–
HDNet-3D	0.184	0.383	0.242	1.706	0.176
DFK-Former (ours)	0.144	0.470	**0.172**	**0.331**	**0.073**

does not guarantee physical fidelity: Curl-SIREN is divergence-free by construction yet exhibits large KE, enstrophy, and spectral errors. The differences become more pronounced on the ten-family benchmark. Divergence-leaking operator methods accumulate substantial enstrophy error, particularly on coherent-structure flows, where derivative-based diagnostics are most sensitive. DFK-Former maintains the lowest enstrophy and spectral errors overall, demonstrating that exact incompressibility can be achieved without sacrificing physically meaningful flow statistics.

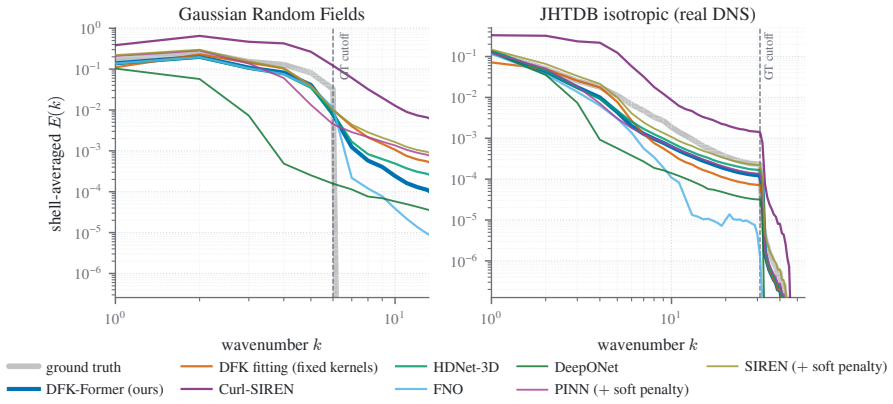

Figure 6: **Energy-spectrum fidelity.** Shell-averaged $E(k)$ (log–log) for ground truth and every method. Both axes are cropped to the band the ground truth carries: the Gaussian Random Fields target is band-limited (Section C.3), so an uncropped axis would compress every curve into the top of the panel. DFK-Former tracks the ground truth across the resolved band.

F.4 Efficiency

Table 19 compares inference cost across methods. DFK-Former reconstructs a scene in 7.5 ms, achieving the highest throughput (134 scenes/s) while requiring the least memory (68 MB). At $M \approx 2$k observations and $Q = 256$ queries, a forward pass requires 41 GFLOPs, approximately half the cost of fixed-kernel DFK fitting (81 GFLOPs) and substantially less than optimization-based alternatives. For this experiment, FLOPs were measured with `torch.utils.flop_counter` (2×MAC convention) for network-based methods and computed analytically for fixed-kernel DFK fitting.

The computational profile is dominated by the transformer encoder, which accounts for more than 99.9% of the total FLOPs; kernel evaluation is comparatively negligible. As a result, increasing the number of kernels has little impact on runtime over the range considered in Section 4. In contrast, optimization-based methods incur a fundamentally different inference cost, solving a reconstruction

problem independently for each field. DFK-Former shifts this cost to training, replacing per-instance optimization with a single feed-forward prediction.

Table 19: **Efficiency battery** Runtime, throughput, peak memory, and computational cost for reconstructing a single scene ($M \approx 2$k, $Q=256$, batch size 1). Lower is better except for throughput.

Method	ms/field ↓	scenes/s ↑	peak mem (MB) ↓	GFLOPs ↓	# params (M)
SIREN / Curl-SIREN (per-field fit)	53000	0.019	204	9500	0.20
DFK fitting (fixed kernels)	22	45	1285	81	0
DFK fitting (adaptive kernels)	9821	0.10	623	786	0
DFK-Former (ours)	**7.5**	**134**	**68**	**41**	10.2

F.5 GENERALIZATION

We evaluate generalization along four axes: unseen sparsity, unseen noise, unseen flow families, and zero-shot transfer to real DNS data. For unseen sparsity and noise, DFK-Former, FNO, and DeepONet are retrained under matched restricted training distributions and evaluated outside the training range (Table 20). For unseen families, training is restricted to Gaussian-random-field and ABC flows, with all remaining coherent flow families withheld (Table 21). Real-DNS transfer requires no additional training, as none of the methods observe JHTDB data during training (Table **??**).

Corruption-level extrapolation. Table 20 shows that DFK-Former generalizes well beyond the sparsity and noise levels observed during training. Restricting training to denser observations increases error by only 0.5% at the most sparse evaluation point ($M = 64$), while restricting training to $\sigma_{\mathrm{rel}} \leq 10\%$ noise increases error by 3.1% when evaluated at 20% noise. Similar behavior is observed for restricted FNO and DeepONet models, indicating that all three architectures are relatively insensitive to moderate extrapolation along these corruption axes. Compared with the matched restricted baselines, DFK-Former consistently achieves the lowest error across sparsity levels and remains the strongest method under severe noise. The relative ranking changes little compared to the full-range models, suggesting that performance differences are not driven by exposure to the test corruption levels during training.

Table 20: **Generalization to unseen sparsity and noise levels.** Restricted models are trained only on moderate sparsity ($M \in [256, 1024]$) or noise ($\sigma_{\mathrm{rel}} \leq 10\%$) and evaluated both within and beyond that range. Columns marked by † correspond to out-of-distribution settings not observed during training. DFK-Former remains competitive under both sparsity and noise extrapolation, with only modest degradation relative to its full-range counterpart. FNO and DeepONet use identical restricted-training protocols. Results report EPE averaged over 24 fields and 3 random seeds. Lower is better.

	Sparsity axis					Noise axis (σ_{rel})			
Method	1% (M=64)†	2% (M=128)†	5%	20%	50%	0%	5%	10%	20%†
Restricted training — sparsity arm $M \in [256, 1024]$, noise arm $\sigma_{\mathrm{rel}} \leq 10\%$, every stage, retrained from step 0									
FNO	0.753	0.665	0.580	0.407	0.362	0.495	0.496	0.497	0.501
DeepONet	0.812	0.782	0.766	0.756	0.750	0.757	0.755	0.748	0.724
DFK-Former (ours)	**0.718**	**0.637**	0.551	0.380	0.323	0.461	0.464	0.471	**0.492**
Full-range training ($M \in [128, 2048]$, $\sigma_{\mathrm{rel}} \leq 20\%$) — unrestricted reference									
FNO	0.742	0.669	0.587	0.405	0.323	0.498	0.498	0.498	0.501
DeepONet	0.816	0.781	0.765	0.753	0.748	0.756	0.753	0.747	0.723
DFK-Former	0.714	0.634	0.542	0.355	0.287	0.447	0.451	0.461	0.478
DFK fitting (fixed kernels, best ridge)	0.804	0.759	0.706	0.546	0.481	0.638	0.638	0.640	0.636
oracle representation floor	*0.053*	*0.051*	*0.047*	*0.048*	*0.049*	*0.049*	*0.048*	*0.052*	*0.041*

Structural generalization is substantially harder. A markedly different trend emerges when entire flow families are withheld during training (Table 21). Removing four of the five coherent-flow families increases error by 4.5–$8.0\times$ on the withheld families and by $2.13\times$ overall. The degradation extends beyond the unseen families themselves: performance on the retained ABC family also deteriorates, whereas the Gaussian Random Fields source distribution remains largely unchanged. These results indicate that coherent-flow diversity contributes directly to the learned prior rather than merely improving interpolation within individual families.

Table 21: **Flow-family generalization under training-set restriction.** EPE on the eleven evaluation sets. The *restricted* model is trained using only the ABC coherent-flow family, while keeping all other training components, sampling weights, corruption ranges, and spectral-flow data identical to the full setup. The ratio reports *restricted/full*. Large ratios on withheld families indicate strong transfer from training on diverse coherent structures, whereas ratios near one indicate limited dependence on flow-family diversity. The JHTDB datasets are zero-shot for both models.

	Set	restricted	full library	ratio
Excluded families	Shear layer	0.425	0.054	**7.95**
	Taylor–Green	0.301	0.053	5.71
	Vortex ring	0.277	0.062	4.50
	Lamb–Oseen	0.249	0.056	4.48
Included families	ABC (the one kept family)	0.074	0.036	2.04
	Gaussian Random Fields	0.369	0.343	**1.08**
Representative mixture	CAF ($4/5$ Excluded families)	0.307	0.101	3.04
Real DNS (zero-shot)	JHTDB channel (wall shear)	0.146	0.063	**2.33**
	JHTDB transitional BL	0.157	0.081	1.93
	JHTDB isotropic (homogeneous)	0.224	0.200	1.12
	JHTDB MHD (homogeneous)	0.272	0.269	**1.01**
	macro (all eleven sets)	0.255	0.120	**2.13**

The effect propagates in a physically meaningful manner. Wall-bounded DNS datasets exhibit the largest degradation, whereas homogeneous turbulence datasets remain comparatively stable. Notably, the restricted model retains the defining properties of the representation, remaining exactly divergence-free and preserving the same inference cost. The loss is therefore attributable to reduced prior coverage rather than a breakdown of the reconstruction mechanism itself.

F.6 TIME-VARYING FLOW: THE FULL BREAKDOWN

Tables 22–24 provide a detailed evaluation on the dynamic JHTDB benchmark (60 flows, 180 test frames), including performance across corruption regimes, DNS datasets, and temporal-model variants.

Temporal priors matter primarily in ill-posed regimes. Table 22 shows that Temporal DFK-Former achieves the lowest error in every under-constrained setting, including contamination, noise, sparsity, occlusion, and their combination. Relative to per-frame DFK fitting, the improvement ranges from $2.6\times$ to $9.7\times$, with the largest gains under sparse and occluded observations. In contrast, the only regime where the temporal model does not achieve the lowest error is `clean&dense`, where observations are sufficiently informative for optimization-based reconstruction to approach the optimal solution within the fixed kernel basis. This contrast highlights the role of the learned temporal prior: its benefits emerge when the reconstruction problem is ambiguous rather than fully determined by the observations. Per-frame DFK fitting additionally exhibits occasional numerical

Table 22: **Time-varying reconstruction of unobserved real DNS frames**, all six corruption regimes (dynamic JHTDB: 60 flows, 180 test observations). Each cell is **median (mean)** endpoint error at times the model never sees. Per-frame DFK's mean is inflated by $15/180$ numerical failures.

Corruption regime	**Temporal DFK-Former**	Per-frame DFK	Shared-kernel DFK	ST-SIREN	ST-Curl-SIREN
`clean&dense`	0.114 (0.114)	0.100 (0.102)	**0.083** (0.093)	0.442 (0.432)	0.781 (0.727)
`contaminated`	**0.085** (0.122)	0.217 (0.291)	0.165 (0.191)	0.365 (0.448)	0.712 (0.781)
`noisy`	**0.088** (0.097)	0.398 (1.874)	0.268 (0.281)	0.440 (0.482)	0.939 (0.926)
`sparse`	**0.109** (0.129)	0.847 (0.828)	1.093 (1.067)	0.388 (0.453)	0.590 (0.617)
`occluded`	**0.098** (0.110)	0.951 (107.1)	1.511 (1.519)	0.413 (0.451)	0.766 (0.819)
`hard combo`	**0.138** (0.173)	0.916 (11.52)	1.420 (1.419)	0.497 (0.528)	1.029 (1.057)
Overall	**0.114** (0.124)	0.767 (20.29)	0.626 (0.762)	0.428 (0.466)	0.842 (0.821)

failures, substantially affecting mean errors in the most challenging settings, whereas all learned temporal models remain stable across the entire benchmark.

Benefits increase with temporal coherence. Table 23 breaks results down by flow type. Improvements over per-frame reconstruction are largest for wall-bounded flows, reaching $14.2\times$ on channel flow and $10.4\times$ on the transitional boundary layer, while homogeneous turbulence datasets show smaller but still substantial gains. These trends are consistent with the amount of exploitable temporal structure available in each dataset. Across all datasets, the predicted fields remain exactly divergence-free up to the numerical precision of the diagnostic.

Table 23: **Temporal reconstruction by flow type.** Median metrics on the dynamic JHTDB benchmark (180 test observations). Temporal DFK-Former consistently outperforms per-frame DFK fitting, with the largest gains on wall-bounded flows. The ratio column reports the relative improvement over a per-frame DFK baseline using the same kernel basis.

Dataset	n_{obs}	EPE	angle (°)	$\|\nabla\cdot\boldsymbol{v}_\Theta\|_{\text{rms}}$	Per-frame DFK	ratio
JHTDB isotropic	48	0.141	6.8	5.6×10^{-7}	0.752	5.3×
JHTDB MHD	48	0.196	9.5	6.3×10^{-7}	0.790	4.0×
JHTDB channel	42	**0.057**	**2.7**	2.1×10^{-7}	0.805	**14.2**×
JHTDB transitional BL	42	0.073	3.4	2.3×10^{-7}	0.753	10.4×

Shared anisotropic kernels provide the strongest temporal model. Table 24 compares alternative temporal parameterizations under a fixed kernel budget. The best-performing configuration uses a shared anisotropic kernel geometry across time, while predicting only time-varying kernel coefficients and mean flow. This design outperforms both per-frame optimization and moving-kernel variants, achieving the lowest overall median error (0.114).

Interestingly, the adaptive kernel placement strategy that improves the static model does not transfer to the temporal setting. Although adaptive placement improves synthetic validation performance, it consistently underperforms fixed shared placement on real DNS data. This suggests that a stable spatial basis shared across time provides a stronger inductive bias than frame-specific geometric adaptation.

Table 24: **Ablation of temporal parameterizations on real DNS sequences.** Median EPE on the dynamic JHTDB benchmark (60 flows, 180 test observations). All temporal variants share the same $K=1024$ kernel budget, training data, and supervision. The best-performing design uses a shared anisotropic kernel geometry with low-order temporal dynamics ($F=2$), outperforming both moving-kernel variants and per-frame DFK fitting. Adding adaptive kernel placement provides no benefit on real DNS data. Lower is better.

Corruption regime	Per-frame DFK	$F=2$ moving iso.	$F=7$ moving iso.	$F=7$ shared aniso.	$F=2$ shared aniso.	+ adaptive placement
`clean&dense`	0.100	0.115	0.174	0.133	**0.114**	0.117
`contaminated`	0.217	0.092	0.172	0.105	**0.085**	0.100
`noisy`	0.398	0.093	0.173	0.102	**0.089**	0.098
`hard combo`	0.916	0.146	0.222	0.152	**0.138**	0.152
`sparse`	0.847	0.118	0.194	0.125	**0.109**	0.117
`occluded`	0.951	0.103	0.186	0.117	**0.098**	0.108
Overall	0.767	0.117	0.183	0.128	**0.114**	0.118
angle / $\|\nabla\cdot\boldsymbol{v}_\Theta\|_{\text{rms}}$	40.0° / $\sim10^{-13}$	5.2° / 6.4×10^{-7}	7.6° / 6.0×10^{-7}	5.7° / 4.9×10^{-7}	**4.8**° / 4.6×10^{-7}	5.0° / 4.7×10^{-7}

Qualitative comparison across observation regimes. Figures 7 and 8 compare representative Forced MHD sequences under two contrasting observation regimes: a well-observed Forced MHD `clean&dense` setting ($M\approx2400$ samples per frame) and a severely under-constrained `sparse` setting ($M\approx420$). In both figures, the top row shows the ground truth and reconstructed velocity magnitude at an unseen time instant, while the bottom row visualizes the corresponding observations and error maps.

In the Forced MHD `clean&dense` regime (Figure 7), optimization-based DFK reconstruction attains the lowest errors. Shared-kernel DFK achieves the best result within the DFK basis, and

per-frame DFK closely matches it, while Temporal DFK-Former reaches comparable accuracy. The resulting error maps are nearly indistinguishable, indicating that the observations are sufficiently informative for optimization alone to recover the best solution supported by the representation. In this regime, the primary advantage of the learned model is computational efficiency rather than reconstruction accuracy.

The behavior changes markedly in the `sparse` regime (Figure 8), where the observations no longer uniquely determine the flow. Both optimization-based DFK variants deteriorate substantially, with errors concentrated in regions far from the observed samples. In contrast, Temporal DFK-Former preserves the dominant flow structures and maintains low reconstruction error despite the reduced observation density. This qualitative trend mirrors the aggregate results in Table 22: between the Forced MHD `clean&dense` and `sparse` regimes, the median error of per-frame DFK increases from 0.100 to 0.847, while Temporal DFK-Former remains essentially unchanged (0.114 to 0.109).

The space-time SIREN baseline exhibits visually plausible reconstructions in some sparse examples, but does not satisfy the incompressibility constraint, exhibiting divergence residuals several orders of magnitude larger than the kernel-based methods. Taken together, these examples illustrate the complementary strengths of optimization and learning: optimization-based fitting is most effective when observations densely constrain the solution, whereas the learned temporal prior provides the largest benefit when reconstruction is under-constrained.

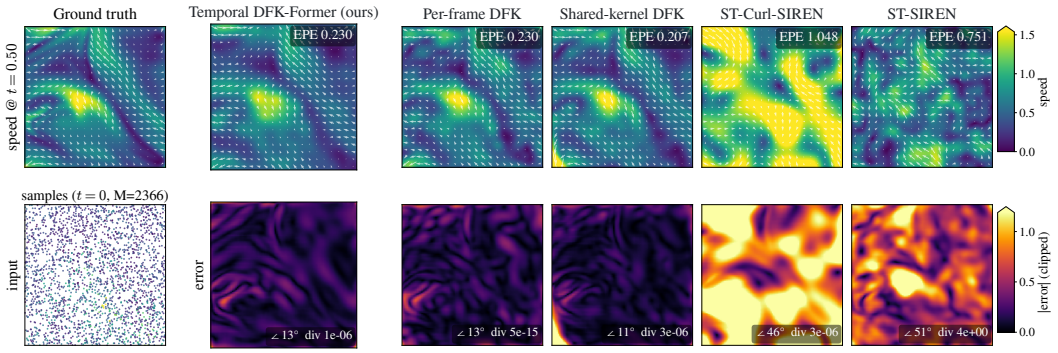

Figure 7: **The well-posed case the model does not win** (Forced MHD (dynamic JHTDB), `clean&dense`): the DFK-span solves reach the fixed-basis optimum — shared-kernel DFK edging the learned prior out (0.207) and per-frame DFK matching it (0.230 each). The space-time INRs remain far behind (0.751, 1.048). The learned prior earns its keep in the ill-posed regimes, not everywhere.

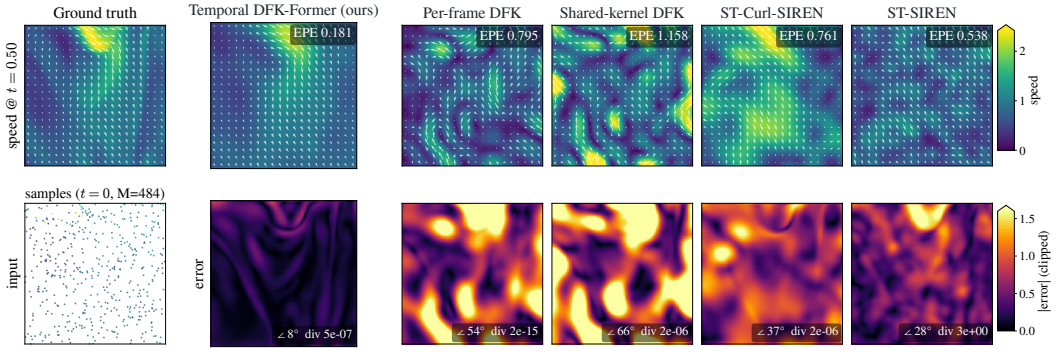

Figure 8: **Temporal reconstruction on a representative real-DNS example** (Forced MHD (dynamic JHTDB), `sparse` regime). Top: ground truth and predictions from Temporal DFK-Former, per-frame DFK fitting, iterative DFK fitting, space-time Curl-SIREN, and space-time SIREN. Bottom: sparse observations and reconstruction errors. DFK-Former best preserves the flow structure and achieves the lowest EPE while remaining exactly divergence-free.

Figure 9 further illustrates representative predictions across time. The model produces a continuous velocity field over t while preserving exact divergence-freeness at unseen intermediate time instances.

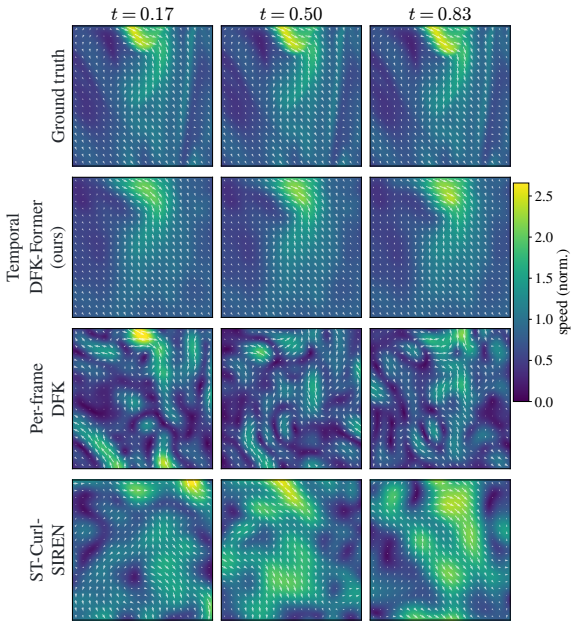

Figure 9: **The reconstruction as a function of time** (Forced MHD (dynamic JHTDB), `sparse` regime). Columns are three real-DNS times the model never observes — frame midpoints; rows are ground truth, our model, and two baselines. The emitted field is continuous in t and exactly divergence-free at every t, not only where it was supervised.

F.7 Real tomographic PIV: 4th International PIV Challenge (Case C)

All previous experiments evaluate synthetic corruptions applied to known velocity fields. In contrast, Case C of the 4th International PIV Challenge evaluates reconstruction from a real tomographic-PIV measurement pipeline. Starting from the released camera images, we perform calibration, volumetric particle reconstruction, and particle pairing, producing a scattered set of particle positions and displacements that serves as input to DFK-Former. The resulting observations therefore include realistic experimental artifacts, including detection noise, ghost particles, and matching errors. As throughout this work, no model is trained or adapted on challenge data.

Benchmark characteristics. Case C consists of a single static displacement field observed under nine seeding-density conditions. The field combines multi-scale vortex structures, regions of nearly uniform displacement, and a deliberately non-solenoidal pseudo-jet component. Reconstructions are evaluated on the challenge grid ($69 \times 229 \times 309$ points at $0.2\,\mathrm{mm}$ spacing). Because the challenge does not provide a reference field, quantitative accuracy cannot be measured directly; evaluation is therefore qualitative. Two aspects are particularly relevant. First, the pseudo-jet region is intentionally not divergence-free, implying that no exactly incompressible representation can recover it perfectly. Second, the absence of ground truth limits analysis to comparisons between reconstructions and measurements rather than reconstruction error relative to the underlying flow.

Observations. Figure 10 indicates that DFK-Former retains substantially more spatial structure than fixed-kernel DFK fitting, particularly at intermediate and small scales. The difference is consistent with the behavior observed on synthetic benchmarks: selecting regularization through sparse measurement consistency favors smoother solutions, whereas the learned prior preserves coherent flow structures. Despite the realistic measurement pipeline, the reconstructed field remains divergence-free up to numerical precision ($\|\nabla \cdot \boldsymbol{v}_\Theta\|_{\mathrm{rms}} = 6.2 \times 10^{-10}$), demonstrating that the representation's physical guarantees remain intact under experimental conditions.

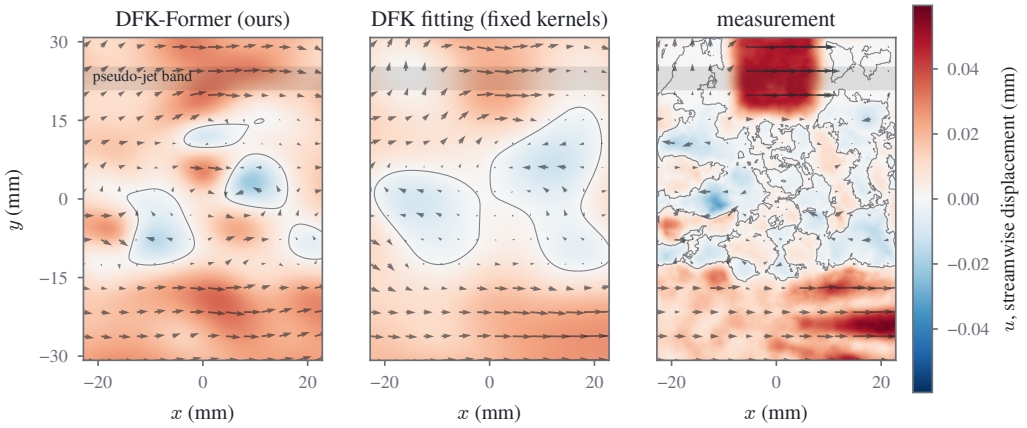

Figure 10: **Real tomographic-PIV reconstruction on Case C of the 4th PIV Challenge.** Mid-plane ($z = 0$) streamwise displacement with in-plane vectors overlaid. Both methods reconstruct from the same sparse (0.01 ppp) particle observations. DFK-Former preserves the dominant vortex structures and coherent displacement patterns visible in the measurement, whereas the fixed-kernel DFK fit produces a substantially smoother field. The shaded region corresponds to the challenge's non-solenoidal pseudo-jet component, which cannot be represented exactly by divergence-free models.

F.8 Toward ground-truthed particle-flow reconstruction

The main limitation of Case C is the absence of a reference solution. The recent 2nd Lagrangian Particle Tracking and Data Assimilation Challenge addresses this limitation by providing particle observations generated from DNS flows and evaluating submitted reconstructions against hidden ground-truth fields. Since benchmark results are not yet available for DFK-Former, we briefly discuss its relevance as a future evaluation platform.

Why this benchmark matters. Unlike Case C, the challenge provides externally evaluated reconstruction metrics, exactly divergence-free target flows, and known particle correspondences. These properties eliminate several ambiguities present in tomographic-PIV evaluation and enable direct measurement of velocity, vorticity, and pressure reconstruction quality. The benchmark includes two complementary configurations. The turbulent square-duct case evaluates reconstruction in a high-Reynolds-number wall-bounded flow with substantial mean transport, directly testing the mean-flow component of the representation. The Rayleigh–Bénard convection case provides a closed-flow configuration dominated by large-scale buoyancy-driven structures and negligible net momentum.

Expected insights. The two datasets also probe different regions of the representation-capacity spectrum. The Rayleigh–Bénard configuration is dominated by large coherent structures that lie comfortably within the scales represented by the current kernel budget. In contrast, the turbulent square-duct flow contains energetic structures at substantially finer scales, making it a useful stress test of the representation's spatial resolution limits. Together, these benchmarks provide a natural next step for evaluating divergence-free reconstruction methods under realistic particle-based sensing conditions.